

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки

Кафедра автоматизації та управління в технічних системах

(повна назва кафедри)

«На правах рукопису»
УДК _____

«До захисту допущено»

Завідувач кафедри

_____ **Ролік О.І.**

(підпис)

(ініціали, прізвище)

“ _____ ” _____ 2018 р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності _____ **8 091401 “Системи управління і автоматизації”**

(код і назва спеціальності)

на тему: Архітектура програмних систем для дистанційного управління

Виконав: студент 6 курсу, групи ІА-61м

(шифр групи)

_____ **Кравінський Василь Олександрович**

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник к.т.н., доц. Катін Павло Юрійович

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант _____

(назва розділу)

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент _____

(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.

Студент _____

(підпис)

Київ – 2018 року

РЕФЕРАТ

Дипломна робота освітньо-кваліфікаційного рівня “магістр” на тему “Архітектура програмних систем для дистанційного управління”: 125 с., 21 рис., 64 таб., 2 додатків, 28 джерел.

Об’єктом дослідження даної роботи є процес розроблення, налагодження і документування складних програмних комплексів управління рухомим об’єктом.

Метою роботи є проведення аналізу існуючих програмних комплексів для дистанційного управління, побудова демонстраційної моделі такого комплексу та його формалізація.

В ході виконання роботи було проведено дослідження існуючих наземних рухомих об’єктів для управління та програмних комплексів для дистанційного управління, виконано їх аналіз та класифікацію. Було розглянуто ключові елементи архітектури таких комплексів та вибрано архітектуру для створення макету комплексу.

Результатом роботи стало створення програмного комплексу для дистанційного управління з використанням управління через веб додаток та побудовано основні діаграми для формалізації комплексу. Отримані результати можуть використовуватись у курсі програмування.

ПРОГРАМНІ КОМПЛЕКСИ ДЛЯ ДИСТАНЦІЙНОГО УПРАВЛІННЯ,
АРХІТЕКТУРА ПРОГРАМНИХ КОМПЛЕКСІВ, ФОРМАЛІЗАЦІЯ, UML,
ДИСТАНЦІЙНЕ УПРАВЛІННЯ.

ABSTRACT

Diploma work of educational qualification level "Master" on the theme "Architecture of software systems for remote control": 125 p., 21 figures, 64 tables, 2 appendices, 28 sources.

The object of study of this work is the process of developing, debugging and documenting complex software complexes for managing a moving object.

The purpose of the work is to conduct an analysis of existing software systems for remote control, construction of a demonstration model of such a complex and its formalization.

In the course of the work, the research of existing ground moving objects for management and software complexes for remote control was carried out, their analysis and classification was performed. The key elements of the architecture of such complexes were considered and architecture was chosen to create the layout of the complex.

The result of the work was the creation of a software complex for remote control using the management through the web application and built the main diagrams for the formalization of the complex. The results obtained can be used in the course of programming.

SOFTWARE COMPLEXES FOR REMOTE CONTROL, ARCHITECTURE OF SOFTWARE COMPLEXES, FORMALIZATION, UML, REMOTE CONTROL.

ЗМІСТ

ПЕРЕЛІК ВИКОРИСТАНИХ СКОРОЧЕНЬ.....	6
ВСТУП.....	7
1 ПРИЗНАЧЕННЯ І СФЕРА ЗАСТОСУВАННЯ	8
2 ОБҐРУНТУВАННЯ ПОТРЕБИ У РОЗРОБЦІ РСДУ НА ОСНОВІ ПКДУ І ОГЛЯД ІСНУЮЧИХ РІШЕНЬ.....	9
2.1 Актуальність використання РСДУ на основі ПКДУ у під час ліквідації наслідків стихійних лих.....	9
2.2 Існуючі технічні рішення – прототипи РСДУ	12
2.2.1 Автоматизовані комплекси на основі РСДУ для рішення комерційних завдань продажу товарів.....	12
2.2.2 Малогабаритні рішення для огляду.....	14
2.2.3 Робототехнічні комплекси на базі РСДУ для надважких умов.....	17
2.2.4 Рішення на базі РСДУ для комунальних служб.....	21
2.3 Класифікація РСДУ.....	22
2.4 Огляд існуючих ПКДУ РО	24
3 ОГЛЯД МОЖЛИВИХ АРХІТЕКТУР ПКДУ РО.....	27
4 ФОРМАЛІЗАЦІЯ ПКДУ РУХОМОГО ОБ’ЄКТУ У ВИГЛЯДІ USE CASE ДІАГРАМ ПРЕЦЕДЕНТІВ.....	29
4.1 Загальний опис.....	29
4.2 Розроблення діаграм прецедентів.....	30
4.2.1 Розроблення загальної діаграми прецедентів ПКДУ.....	32
4.2.2 Розроблення діаграми прецедентів підсистеми об’єкту управління.....	33
4.2.2.1 Програмна компонента рухомого об’єкту	44
4.2.3 Розроблення діаграма прецедентів веб-додатку	45
4.2.3.1 Розроблення Web додатку для віддаленого управління.....	74
4.2.3.2 Графічний інтерфейс користувача.....	77
5 ФОРМАЛІЗАЦІЯ ПОВЕДІНКИ ПКДУ ЗА ДОПОМОГОЮ UML ДІАГРАМ.....	81

5.1 Математична формалізація технічної системи у вигляді FSM.....	82
5.2 UML Діаграма станів «Сайту для віддаленого управління»	86
5.3 UML Діаграма станів «Рухомого об'єкту»	88
5.4 Розроблення діаграм послідовностей.....	89
5.4.1 Розроблення діаграми послідовності (загальної).....	90
5.4.2 Розроблення докладної діаграми послідовності	92
6 РОЗРОБЛЕННЯ СТАРАПІ ПРОЕКТУ	93
6.1 Опис ідеї проекту	93
6.2 Технологічний аудит ідеї проекту	94
6.3 Аналіз ринкових можливостей запуску стартап-проекту	96
6.4 Розроблення ринкової стратегії проекту.....	103
6.5 Розроблення маркетингової програми стартап-проекту	107
ВИСНОВОК.....	111
СПИСОК ЛІТЕРАТУРИ.....	112
ДОДАТОК А Лістинги основних класів сайту для віддаленого управління	115
ДОДАТОК Б Публікації	118

ПЕРЕЛІК ВИКОРИСТАНИХ СКОРОЧЕНЬ

АРП – автоматизована рухома платформа

ВБУ – віддалений блок управління

НРО – наземний рухомий об'єкт

ПЗ – програмне забезпечення

ПКДУ – програмні комплекси для дистанційного управління

РО – рухомий об'єкт

РСДУ – рухомі системи дистанційного управління

ВСТУП

Архітектура програмних систем для дистанційного управління, створення на її основі програмних комплексів для дистанційного управління та їх формалізація широко використовуються у багатьох сучасних світових галузях. Розглянуті аналоги не мають достатньої функціональності для вирішення необхідних завдань, тому запропонований варіант реалізації є актуальним не тільки в Україні але і у світі загалом.

Вивчення робототехніки є невід’ємною складовою навчального процесу кафедри АУТС, тому дослідження виконані у даній дисертації будуть корисними для вивчення та застосування. Також створений програмний макет можна використовувати у вивченні програмування.

Метою даної роботи є побудова наочної моделі ПКДУ і її формалізація. Для досягнення цієї мети буде проведено огляд існуючих рішень РО і ПКДУ. На основі огляду рішень буде формалізовано власний ПКДУ та створено його макет.

Аналіз наукових робіт і літератури [2-7], показує що формалізація ПКДУ, є досить складною, і існуючі рішення громіздкі, тому у даній роботі запропоновано новий варіант формалізації з використанням посилань.

Практичне застосування дисертації полягає у тому, що розроблений макет ПКДУ можна використовувати при вивченні курсу «Програмування», а також при вивченні робототехніки.

Конференції.

Публікації.

1 ПРИЗНАЧЕННЯ І СФЕРА ЗАСТОСУВАННЯ

Протягом останніх років необхідність різних сфер, як соціальних так і економічних, у РСДУ значно зростає.

Рухливі роботи застосовуються переважно в екстремальних умовах, коли люди не можуть перебувати всередині транспортного засобу або якщо їх перебування там пов'язане з ризиком для життя. Крім того, в умовах підвищеної небезпеки людина починає робити помилки, знижується його працездатність і ефективність дій.

Роботи здатні замінити людей при проведенні аварійно-рятувальних робіт в атомній енергетиці, в хімічній, нафтогазовій і гірничодобувній галузях; під час ліквідації наслідків стихійних лих, в операціях по боротьбі з тероризмом, при охороні об'єктів і патрулювання територій.

У ряді випадків доцільно використовувати роботи, дистанційно керовані оператором або групою операторів по теле- і радіоканалах зв'язку. Однак майже завжди необхідно забезпечити короткочасне, але повністю автономне функціонування робота (в першу чергу його системи управління рухом) на випадок раптового припинення радіозв'язку при входженні в зону відсутності сигналу. Крім того, телеуправління транспортним роботом вимагає від оператора постійної уваги, що вельми втомлює. У деяких робочих середовищах (зокрема, в тумані, в диму, вночі), де оператори часто втрачають орієнтування.

Розглянуті у даній роботі ПКДУ можна використовувати у будь-якій з розглянутих сфер застосування.

2 ОБҐРУНТУВАННЯ ПОТРЕБИ У РОЗРОБЦІ РСДУ НА ОСНОВІ ПКДУ І ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

2.1 Актуальність використання РСДУ на основі ПКДУ у під час ліквідації наслідків стихійних лих

Найбільшої необхідності РСДУ на основі ПКДУ набувають у місцях, де не присутність людини не просто небезпечна, а неможлива, оскільки це може спричинити швидку смерть. Такими місцями є зони ліквідації наслідків аварій на атомних електростанціях. Розглянемо дві найвідоміші аварії.

Через те, що аварія на Чорнобильській АЕС мала колосальні масштаби, до ліквідації наслідків були залучені кращі фахівці колишнього Радянського Союзу. Спеціальна військова і будівельна техніка, роботи, а також спеціальний автотранспорт, пристосований для роботи в умовах сильного радіоактивного опромінення, були широко задіяні при влаштуванні плити під зруйнованим реактором, очищення даху 4-го блоку Чорнобильської АЕС, при похованні «рудого лісу», при дезактивації території ближньої зони ЧАЕС, а також при посадці лісу і трав на території ближньої зони.

Пристрій плити під зруйнованим реактором. Масштаб катастрофи став очевидним вже в перші дні після аварії, тому фахівці висловили побоювання, що нижній ярус будівельних конструкцій може не витримати доданих до неї температурних навантажень і додаткового тиску, який виник від насипаних вертольотами зверху 5000 тонн матеріалів. Вони попередили, що якщо паливо провалиться вниз, то неминуче викличе серйозне забруднення ґрунтових вод. Виникла необхідність створення якогось бар'єру, який міг би перепинити шлях руху розпеченого палива з розплавленого ядерного реактора в ґрунтові води. Вирішили побудувати величезну залізобетонну плиту під зруйнованим реактором 4-го блоку Чорнобильської АЕС. Причому плита ця повинна була стати унікальною - з системою трубопроводів всередині для подачі води і постійного охолодження простору під реактором. Тут же повинні були бути змонтовані вимірювальні прилади різного призначення. Вже 3 травня 1986 роки роботи почалися, в яких в

цілому взяли участь 388 шахтарів, які приїхали з московського вугільного басейну і з Донбасу. Під фундаментом 4-го енергоблоку проклали тунель довжиною в 136 метрів і діаметром 1,8 метра, проклали з цього тунелю залізничні рейки і необхідні комунікації. З-під плити реактора вибрали весь ґрунт і поклали арматуру для подальшого бетонування. Ґрунт вивозили вручну в вагонетках до котловану.

Очищення даху Чорнобильської АЕС. Коли на 4-му блоці ЧАЕС пролунали два потужні вибухи, високоактивні фрагменти активної зони реактора злетіли в повітря і, описавши в повітрі дугу і зробивши кілька сальто в польоті, звалилися на дах 3-го блоку. Ці шматки створювали вкрай несприятливі умови для спорудження «укриття» над зруйнованим реактором, так як створювали на даху радіоактивне випромінювання дуже високої потужності і представляли смертельну небезпеку. У зв'язку з цим було прийнято рішення провести очищення покрівлі від настільки небезпечних об'єктів. Для цього було підготовлено спеціальне технічне рішення, яке включало в себе видалення залишків руберойду і високорадіоактивних уламків механічним способом, а також нанесення на очищену покрівлю спеціального ізолюючого покриття. У розробці технічного регламенту для проведення цих робіт брали участь ВНДІ АЕС та Інститут атомної енергії ім. Курчатова [8].

Такі ж заходи були прийняті після аварії на АЕС у місті Фукусіма Японія.

Аварія на АЕС «Фукусіма-1» викликана не тільки через землетрус, факти говорять, що сама станція досить успішно витримала сейсмічні поштовхи. Однак проблема була в тому, що тут сталося накладення двох стихійних лих, що і призвело до такої масштабної катастрофи. Хоча офіційне розслідування причин аварії ще не завершено - її висновки будуть готові тільки до кінця року, попередні висновки показують, що землетрус був причиною втрати зовнішнього енергопостачання. Після цього, як і належало, було запущено дизель - генератори, але їх робота порушилася через хвилі цунамі.

Відразу після вибуху стався сильний підвищення рівня радіації, яка досягла понад 1000 мкЗв / год, але через кілька годин, рівень радіації впав до 70,5 мкЗв / год. Пересувні лабораторії, які взяли проби на території АЕС, показали наявність цезію, що могло вказувати на порушення герметичності оболонок

тепловиділяючих елементів. Уряд Японії в полудень цього ж дня, підтвердило, що дійсно стався витік радіації, але про масштаби не повідомлялося. Згодом, офіційні особи, як з уряду, так і з компанії ТЕРСО, в чиєму веденні знаходиться АЕС, заявили, що для охолодження реактора, в його гермооболонку закачуватиметься морська вода, змішана з борною кислотою, а за деякими відомостями, воду будуть закачувати і в сам реактор. За офіційною версією, водень просочився в простір між сталеву оболонку і бетонною стіною, там змішавшись з повітрям, він і вибухнув.

На наступний день, на АЕС «Фукусіма-1» почалися проблеми з блоком № 3. У нього виявилася пошкодженою система аварійного охолодження, яка повинна була підключитися при зниженні рівня теплоносія нижче заданого. Так само, попередні дані говорили, що тепловиділяючі елементи частково оголилися, тому знову виникла загроза вибуху водню. Почався контрольований скид пара з герметичної оболонки, для зниження тиску. Так як не було можливості охолодження реактора блоку №3 в нього, теж, почали закачування морської води.

Однак, вжиті заходи, не допомогли уникнути вибуху на третьому енергоблоці. Вранці 14 березня на цьому блоці прогрімів вибух аналогічний вибуху на першому енергоблоці. При цьому і корпус реактора, і герметична оболонка не постраждали. Персонал став відновлювати аварійне енергопостачання на 1 і 2 блоках, а підкачка морської води здійснювалася на 1 і 3 блоки. Надалі, в цей день відмовила система аварійного охолодження і на другому енергоблоці. ТЕРСО повідомила, що на цьому блоці приймаються такі ж заходи, як на 1 і 3 блоках. Під час закачування морської води в 2 блок, відмовив запобіжний клапан для скидання пари, тиск зріс, і закачування води стала неможливою. Через тимчасового повного оголення активної зони, частина тепловиділяючих елементів ушкодилася, але згодом вдалося відновити функцію клапана, і відновити подачу морської води.

Як видно з наведених хронологій подій, та заходів, спрямованих на ліквідацію наслідків – через високий рівень радіаційного фону люди не були в змозі, без наслідків для здоров'я, виконувати певні роботи [9].

Тому у певній мірі на ЧАЕС і на Фукусімі використовували спеціальні РСДУ.

Оскільки в наш час кількість атомних електростанцій на планеті досить велика, і така катастрофа може статися будь якої миті – необхідність у розвитку РСДУ досить велика.

2.2 Існуючі технічні рішення – прототипи РСДУ

2.2.1 Автоматизовані комплекси на основі РСДУ для рішення комерційних завдань продажу товарів

Першим прикладом, що може бути прототип РСДУ є автономний пристрій фірми Starship Technologies [10], що призначений для доставки малогабаритних легких вантажів по загальних шляхах доставки. Така рухома автоматична платформа уже використовується у США у штаті Вірджинія, вона зарекомендувала себе як легкий та дешевий спосіб доставки вантажів.

Компанія Starship Technologies - європейська компанія, зробила революцію в екологічній і чистій доставці, з автоматизованою системою управління роботом, для переміщення необхідного купленого товару до покупця.

Їх роботи є легкими і недорогими, що дозволяє компанії скоротити поточні витрати на доставку до 15 разів. Роботи рухаються автономно більшу частину часу, але контролюються пультом дистанційного керування, який може взяти контроль у будь-який час.

Постачання останніх кілометрів є найбільшим складним етапом у галузі доставки, особливо місцевими постачаннями, такими як продукти харчування та продукти харчування. Основні характеристики РО [10]:

- зона покриття до 4 кілометрів;
- період виконання доставки 20-40 хвилин,
- вага робота 18 кг,
- вантажопідйомність 8 кг;
- швидкість переміщення 7 км/г.

Робот рис.1 може рухатись у автоматичному та автоматизованому режимах, оператор при необхідності може перемикає режими. Енергопостачання даного

робота забезпечується літій-іонними акумуляторними батареями. За законами штату, де він широко використовується, необхідно проводити нагляд за роботом, в процесі його переміщення. У випадку відхилення від маршруту, або надзвичайному випадку – оператор повинен провести коригування маршруту або вимкнути робота. Пристрій має виконувати правила переміщення по дорозі, слідкувати за рухом пішоходів, для уникнення наїздів, та мати змогу розпізнавання дорожніх знаків. Усі створені апарати мають маркування (аналог номера автомобіля) для ідентифікації, необхідну інформацію про власника, для змоги зв'язку, також апарати повинні мати загальний ідентифікаційний номер (аналог номера кузова автомобіля). Забороняється транспортувати вантажі, які можуть нести загрозу для оточення та людей.

Даний апарат також буде досить корисним у туристичній галузі, для допомоги туристам, доставки їм необхідних товарів.



Рисунок 1 - Автоматизована рухома платформа Starship Technologies [10]

Другим яскравим прикладом такого роду РО може бути робот Dispatch[11].

Результатом стартапу, розробленого у Сан Франциско, США, є робот для доставки Dispatch. На даний момент він ще не набув такої широкої популярності як попередній РО, проте він стає все більш популярним у регіоні розробки.

Головними перевагами Dispatch рис. 2 є більший радіус покриття, досконаліша система автоматичної роботи, більша вага перевезення та довша автономність.

При розробці автоматичної системи управління було застосовано нейронні мережі, з можливістю навчання. Тобто даний робот може вдосконалювати навички керування, розпізнавати більше дорожніх знаків, краще знаходити вихід з нештатних ситуацій.



Рисунок 2 - Робот для доставки Dispatch [11]

2.2.2 Малогабаритні рішення для огляду

Першим прикладом малогабаритного рішення роботу для огляду є робот Riley створений Австралійською компанією iPatrol [12].

Дане рішення представляє з себе невелику гусеничну платформу, здатну самостійно рухатися по дому, знімаючи обстановку на HD-камеру. Зрозуміло, що вона не оснащена засобами та не спрямована на те, щоб забезпечити протидію небажаним гостям.

Камера з роздільною здатністю 5 Мп здатна працювати у відсутності денного освітлення, крім того, вона фіксує рух, та повідомляє власника шляхом надсилання повідомлення в мобільному додатку.

Платформа легко справляється з порогами та іншими перепадами висоти підлоги в різних приміщеннях будинку, але сходи являються для неї нездоланною перешкодою.

Пристрій рис. 3 оснащений мікрофоном і динаміками – тому з людьми у будинку можна спілкуватися. Також даний засіб можна використати для гри з домашніми улюбленцями і слідкуванням за ними.



Рисунок 3 - Робот для огляду Riley [12]

Другим варіантом прототипу автоматизованої платформи є апарат SpyRobot 4WD від фірми Macroswiss, що має 4 спеціальної форми [13]. Зовнішній вигляд робота показаний на рис.4. Особливістю цього робота є його колеса, що дають йому можливість проїзду через важкодоступну місцевість. Форма коліс реалізована таким чином, що під час руху, платформа «вгризається» у поверхню, по якій пересувається. Також, SpyRobot має змогу пересуватися по воді, колеса працюють як колесо катамарану, що дозволяє йому розвивати досить високу швидкість. Оскільки матеріал коліс пружний, то при русі по пересічній місцевості, колеса працюють також як підвіска, зменшуючи вібрації. Міцна конструкція платформи, дає змогу кидати її на відстань 10 метрів без будь яких наслідків для неї.

Технічні характеристики [13]:

- вага платформи 5 кг;
- максимальна швидкість руху 12 км/г;

- максимальний кут підйому 45 градусів;
- радіус дії до 500 м.

Особливість конструкції платформи, дозволяє їй рухатись і у перевернутому стані. Наявність камери та фари у передній частині роботу дозволяє використовувати його навіть у темну пору доби.



Рисунок 4 - Зовнішній вигляд SpyRobot 4WD [13]

Також є шестиколісна версія такої платформи рис. 5. Габарити і вага модифікованої версії трішки більша, проте прохідність і функціональність вища.

Розробляється декілька версій удосконаленого 6x6 РО. Передбачена військова версія. Військова версія буде оснащена різними видами озброєння, мати додаткові сенсори, для отримання більшої інформації з поля бою, також така версія буде мати підсилений корпус для військових дій.

Також у розробці версія для міського використання. Така версія буде легшою, менш функціональною, проте її ціна буде значно нижчою. Загальна функціональність не буде змінена.

Оскільки основною перевагою даних платформ (як 4x4 так і 6x6) є прохідність – то вони будуть незамінні для різних цілей у містах, оскільки там усебіч наявні різні перешкоди, важкодоступні місця.

Вага даних платформ дозволяє переносити їх одній людині, що значно розширює сфери застосування, практичність і знижує вимоги до них.



Рисунок 5 - Зовнішній вигляд SpyRobot 4WD 6x6 [14]

2.2.3 Робототехнічні комплекси на базі РСДУ для надважких умов

Першим прикладом, що може бути прототипом РСДУ для надважких умов являється робот для вакуумної дезактивації, розроблений компанією Toshiba [15], який направлений на дистанційне дослідження і дезактивацію радіаційно-небезпечних ділянок атомної електростанції Фукусіма Даїчі (Fukushima Daiichi). Необхідність розробки і використання такого пристрою пояснюється вкрай утрудненим доступом персоналу в окремі приміщення зруйнованих енергоблоків - поверхні (стіни, підлогу і обладнання) забруднені радіоактивними речовинами. В таких умовах високий радіаційний фон ускладнює роботу персоналу з вивчення наслідків катастрофи і вивчення стану будівельних конструкцій та обладнання станції.

Представлений робот рис.6 здатний дезактивувати високо забруднені ділянки підлоги і стін приміщень АЕС Фукусіма за допомогою новаторського методу очищення. Спочатку цей метод вакуумного очищення розроблявся для очищення поверхонь літаків від фарби, але його висока ефективність дозволила застосувати його до завдань радіаційної дезактивації аварійної АЕС.

Метод вакуумної дезактивації реалізованих в роботі полягає в використанні сухого льоду, який спочатку розкидається на забруднене радіоактивними речовинами місце. Інтенсивне випаровування сухого льоду приводить до видалення радіоактивних речовин, які піднімаються в повітря і відразу засмоктуються пилососом встановленим на роботі. Оскільки цей робот був призначений для авіабудівників, для роботи в умовах зруйнованої АЕС робот обладнали гусеницями і чотирма камерами відеоспостереження. Така модернізація робота дозволить дистанційно керувати рухом і роботою робота з очищення.

На даний момент робот здатний очищати 22 квадратних метра на годину забрудненої поверхні. У той же час, запас сухого льоду в роботі тільки на півтори години роботи. Це означає, що при роботі в приміщеннях аварійної АЕС через півтори години необхідно або витягувати робота для заправки сухим льодом в безпечних умовах або підносити додатковий сухий лід вручну.

Ще одним прикладом таких платформ є Dragon Runner показаний на рис.7. Робот розроблений фірмою Qinetiq [16]. Основне призначення робота – це робота у замкнених просторах та маленьких приміщеннях, куди людині важко добратися, а застосування великогабаритних роботів неможливе.



Рисунок 6 - Автоматизована рухома платформа Toshiba [15]

Крім того, за рахунок його конструкції, він може виконувати роботу у місцях, небезпечних для людей. Дана платформа обладнена 4 камерами, спрямованими на відображення відео у різних діапазонах. Передача відео можлива на відстань до 1км.



Рисунок 7 - Автоматизована рухома платформа Toshiba [16]

Робот Precision Urban Hopper [17] має невеликі розміри, компактний, пересувається на 4 колесах. Але, його відокремлює від інших подібних роботів змога долати перешкоди шляхом перепригування.

Для реалізації розглянутої функції, у нього наявна спеціальна пневматична нога. При необхідності, оператор може надіслати сигнал для її використання.

Така особливість дозволяє РО не тільки перепригувати перешкоди, але, при необхідності, перекидатися, відштовхувати інші об'єкти.

Платформа розрахована на подолання перешкод заввишки до 10м. Попередній зовнішній вигляд РО зображено на рис.8. За останньою інформацією, прототип такого роботу уже в змозі підпригувати на висоту, що перевищує його власні розміри у 40 разів.

Конструкція роботу забезпечую достатню міцність для падіння зі значно більшої висоти, ніж він може пригати, тому сфери застосування таких роботів досить широкі.



Рисунок 8 - Зовнішній вигляд Precision Urban Hopper [17]

2.2.4 Рішення на базі РСДУ для комунальних служб

РСДУ також використовуються цивільними службами, для спрощення робіт у важкодоступних місцях.

Для прикладу розглянемо робота для огляду труб [18] рис. 9. Такий робот використовується комунальними службами різних країн для огляду труб водопостачання, або каналізаційних труб для виявлення проблемних ділянок.

Робот обладнаний камерою та підсвіткою. Передача відео та електроенергії відбувається через кабель. Максимальна заявлена довжина якого до 200м.



Рисунок 9 – Робот для огляду труб [18]

Наступним прикладом РСДУ для комунальних служб є робот для діагностики лінії електропередачі Explorer рис. 10, розроблений Японською компанією HiBot [19].

Такий робот використовується для огляду ліній електропередачі для виявлення пошкоджень проводів, їх корозії та деформацій.

Його особливістю є те, що він рухається по проводах, як по рейках. В його конструкції наявні спеціальні лазерні датчики, які дозволяють знаходити місця ржавіння на проводах. При виявленні пошкодження – оператору надсилається

відповідна інформація, у яку вміщуються географічні координати місця пошкодження.

Також присутні 8 камер, які дозволяють детально оглянути практично будь яку ділянку лінії. Після проходження роботом усієї лінії, електрики точно знають усі місця лінії електропередачі, у яких необхідно виконати роботу.

Крім того, робот може долати різні перешкоди, які можуть трапитись на його шляху, для чого у його конструкції передбачений зміщений центр ваги.



Рисунок 10 – Робот для діагностики лінії електропередачі Expliner [19]

2.3 Класифікація РСДУ

Отже, в результаті проведеного огляду РСДУ різного призначення, видів, типів управління, можна узагальнити інформацію і описати класифікацію РСДУ більш докладно:

РСДУ за типами управління можна розділити на 4 групи:

- провідні з віддаленим управлінням;
- безпроводні з віддаленим управлінням;
- провідні без віддаленого управління;
- безпроводні без віддаленого управління;

Більшість розглянутих РСДУ були провідними без віддаленого управління або безпровідними без віддаленого управління.

Перші дві групи найпоширеніші, оскільки їх реалізація є дешевшою, не потребує додаткових налаштувань та менш вимоглива. Проте наявність віддаленого управління може значно спростити використання РСДУ у майбутньому.

Третя і четверта групи є досить коштовними у виробництві та підтримці, їх налаштування являє собою досить важкий процес по синхронізації обладнання, створенню додаткових шляхів передачі даних, забезпечення безпеки доступу до РСДУ.

За масою їх також можна розділити на 4 основні групи:

- надлегкі, масою до 10 кг;
- легкі, масою до 40 кг;
- середні, масою до 80 кг;
- важкі, масою понад 200 кг.

У світі найбільш поширеними стали РСДУ перших трьох груп. Проведений огляд також показав, що для описаних цілей найкраще використовувати перші дві групи. Такий вибір можна пояснити перш за все простотою у переміщенні таких РСДУ, їх ціною, маневреністю, конструктивними особливостями, що дозволяють їм проникати у важкодоступні місця, прохідністю а також адаптивністю.

На разі ходова частина майже усіх аналогів РСДУ реалізована без використання у підвісці амортизування. Деякі пристрої вирішують дану проблему з використанням модифікованих коліс. Перш за все це використовується у пристроях, які спрямовані на роботу у будівлях та у важкодоступних місцях.

Управління швидкістю руху відбувається головним чином завдяки зміні відносної частоти обертання колі платформи. Більшість малогабаритних аналогів РСДУ характеризуються відносно невисокою швидкістю, тому точність регулювання швидкості – необхідний критерій при реалізації таких платформ.

У загальному випадку РСДУ включає в себе керувально-інформаційну складову(засоби для управління, необхідні датчики, систему відображення відео),

що розташована безпосередньо на РО; командний пункт для оператора РСДУ (програмні та технічні засоби для керування і отримання інформації з платформи); засоби для передачі інформації між РО та операторським технічним забезпеченням.

Наразі будь яка РСДУ у певній мірі потребує втручання з боку людини. Ступінь втручання людини визначає складність реалізації управління та резервного програмного забезпечення, що використовується у РСДУ. Найсучасніші системи зводять втручання людини у діяльність РО до мінімуму, тобто оператор тільки встановлює необхідні завдання, які повинна виконати РСДУ і запускає її.

Таким чином, узагальнюючи аналіз можна визначити вимоги РСДУ у цілому відповідають характеристикам прототипів. Система автоматизованого управління РСДУ має складатися з трьох базових елементів, а саме:

- програмних і апаратних елементів автоматизованої системи управління розташованих безпосередньо на борту РСДУ;
- системи зв'язку, що дозволяє надійно забезпечувати зв'язок між елементами автоматизованого управління;
- програмних і апаратних елементів пульта управління.

Не дивлячись на достатню кількість прототипів можна зробити висновок, що, безпілотні і дистанційно керовані РСДУ це актуальний науковий напрямок, що потребує продовження практичних та теоретичних досліджень.

2.4 Огляд існуючих ПКДУ РО

Прешим прикладом ПКДУ РО є Webot від фірми Wicron рис. 11. Webot - робот телеприсутності, що дозволяє людині здійснювати дії в місці пошуку роботи, використовуючи комп'ютер і Інтернет. Він дозволяє дистанційно спостерігати за подіями і спілкуватися з людьми, бачити навколишній простір і спокійно рухатися по ньому зі швидкістю людської ходьби[20].

Схема робота аналогічна людині. В неї є голова з вбудованою відеокамерою і мікрофонами які будуть вашими вухами і очима. Вони встановлені на поворотній платформі, що дає змогу імітувати рухи голови, подібно до людей. Завдяки новим

технологіям в області телеконференцій якість звуку та відео обмежується тільки пропускною здатністю вашого інтернет каналу і динамічно підлаштовується під нього. Використання абсолютно нового кодека істотно знижує обсяг трафіку.



Рисунок 11 – Робот присутності Webot [20]

Також він має досить широкі області застосування. Віддалені консультації в торговому центрі, допоможуть відвідувачу в пошуку товару. Система дозволяє зменшити кількість консультантів у великому торговому центрі за рахунок використання функцій робота. Здійснювати контроль за дисципліною співробітників, відвідуваність робочих місць, спілкування з співробітниками. Директор завжди може бути присутнім на нарадах. Окрім цього, за допомогою робота і веб додатку гості завжди можуть відвідати офіс.

Окрім цього, на сайті даного робота наявне дистанційне управління, інтерфейс якого зображено на рис. 12.

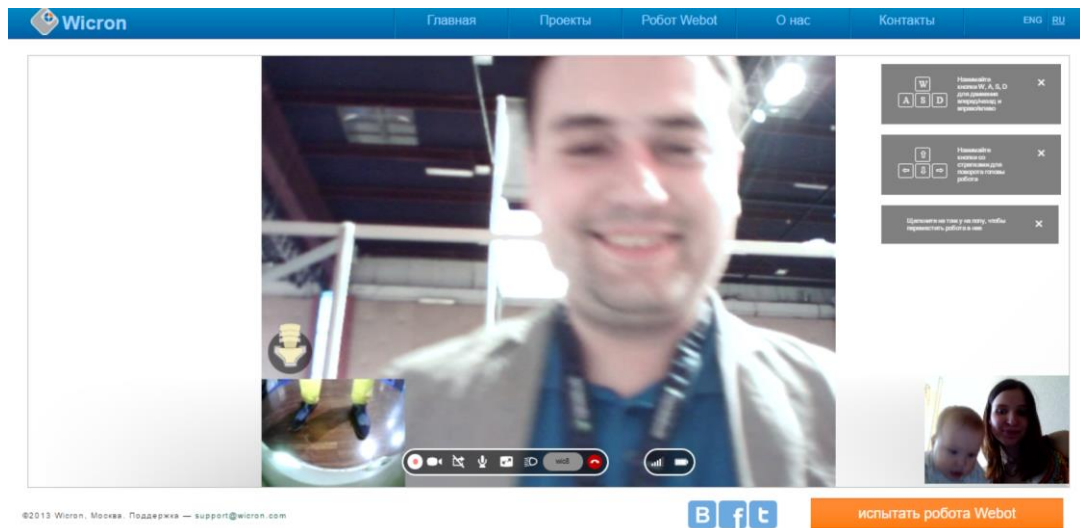


Рисунок 12 – Веб додаток для дистанційного управління Webot [20]

Даний сайт досить корисний, при реалізації ПКДУ, оскільки він дає змогу без встановлення додаткового ПЗ віддалено керувати РО.

Ще одним прикладом ПКДУ РО можна розглянути ПКДУ розроблений для робота на основі RASPBERRY PI [21], веб інтерфейс для управління яким зображено на рис. 13.

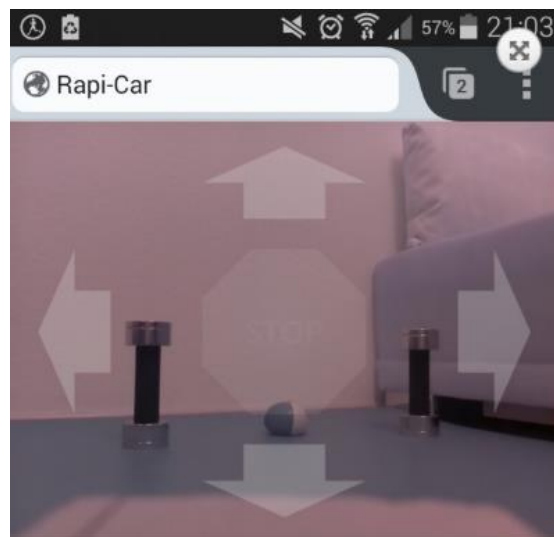


Рисунок 12 – Веб додаток для дистанційного управління Webot [21]

Зв'язок між ВБУ та РО у даному ПКДУ реалізований на базі WIFI з'єднання. Сайт представляє з себе просту HTML сторінку, у якій використано скрипти для управління зі спеціальної Python бібліотеки.

Нажаль, ПКДУ у складі яких наявний веб інтерфейс для віддаленого управління дуже мало. А тих, у яких є доступ для огляду ще менше.

З розглянутих ПКДУ, можна зробити висновок, що навіть наявні у ВБУ, у яких наявні сайти для дистанційного управління шляхом використання веб інтерфейсу, є досить вузькими у використанні, і не забезпечують достатньої функціональності для загального використання. Тому необхідність у дослідження та розробці таких ПКДУ досить висока.

3 ОГЛЯД МОЖЛИВИХ АРХІТЕКТУР ПКДУ РО

При створенні ПКДУ потрібно дотриматись усіх вимог створення таких типів архітектури.

Використання різних програмних та апаратних рішень дозволяють створити велику варіацію можливих архітектур ПКДУ. Розглянемо які основні та додаткові елементи може містити така система.

Першим і основним елементом являється програмне та апаратне забезпечення РО. В залежності від ціни та функціональності апаратного забезпечення рухомого об'єкту, залежить необхідність створення посередників між сайтом для управління та РО.

Другим важливим елементом є сайт для віддаленого управління та тип серверу на якому він розміщений. Можна використовувати різні варіанти створення ПКДУ, в залежності від яких варто вибирати ту чи іншу технологію для реалізації сайту та вибору архітектури, що використовується на сервері.

Третім елементом є ПЗ та технічне забезпечення оператора. Якщо технічне забезпечення РО не дозволяє вести пряму взаємодію з сайтом для віддаленого управління, виникає можливість у використанні пункту управління для оператора. Він використовується у ролі посередника між РО та сайтом для віддаленого управління.

Важливими компонентами архітектури також є шляхи передачі даних між основними елементами. Для зв'язку між сайтом для віддаленого управління та РО або пунктом управління для оператора, та пунктом управління для оператора та РО.

Спираючись на результати огляду існуючих рішень РО та розглянутих ПКДУ нами була вибрана наступна реалізація для ПКДУ.

У якості технології реалізації сайту для віддаленого управління було вибрано ASP.NET Core, оскільки дана технологія є новітньою і найкраще підходить для реалізації такого типу web-додатку. В ролі бази даних для сайту буде використано MySQL.

Розроблений макет буде розміщено на локальному ПС, оскільки він задовольняє усі необхідні вимоги по швидкодії.

Обмін даними між сайтом для віддаленого управління та рухомим об'єктом буде здійснюватись через мережу інтернет та з використанням HTTP.

ПЗ для оператора буде реалізовано у демо варіанті, що буде імітувати дії РО.

Для забезпечення безпеки, буде використано реєстрацію на сайті на основі Access Token, що забезпечить достатній рівень безпеки.

4 ФОРМАЛІЗАЦІЯ ПКДУ РУХОМОГО ОБ'ЄКТУ У ВИГЛЯДІ USE CASE ДІАГРАМ ПРЕЦЕДЕНТІВ

4.1 Загальний опис

Для аналітичного супроводження буде використано UML діаграми[1], що є одним з найбільш актуальних на даний час видів в даний час способів візуалізації та моделювання ПЗ є UML. Діаграми UML були створені для візуалізації, проектування, супроводження та документування, програмних систем. Також діаграми UML можуть бути використані для формалізації інших технічних систем, економічних систем, бізнес процесів, системного проектування, відображення організаційних структур, тощо.

Для розробки, налагодження і супроводження ПКДУ рухомими об'єктами потрібна загальна формалізація програмного забезпечення (ПЗ). Формалізація для ПКДУ ускладнюється з наступних причин

- через використання різних технологій ПЗ, що застосовуються в ПКДУ;
- у ПКДУ поєднуються високотехнологічні елементи ПЗ пульта управління з графічними інтерфейсами і ПЗ виконавчих механізмів рухомих об'єктів. Отже, з'являється потреба формалізації декількох парадигм (ООП, процедурний підхід, функціональна парадигма) створення ПЗ у одній програмній системі РСДУ РО;
- проблема синхронізації ПКДУ у єдиній системі, зазвичай оформленого як набір текстових файлів, і його візуального відображення;

Особливо ці претензії можна пред'явити до ранніх поколінням UML.

Крім того уніфікована мова моделювання дає можливість розробникам ПЗ отримати узагальнення в графічній інтерпретації ПКДУ, описуючи всі його аспекти, послідовність дій, множину станів, принципи роботи, тощо.

Для цього, не дивлячись на розглянуті недоліки, UML має достатньо можливостей, що дозволяють забезпечити формалізацію ПКДУ у одному опису. До цих можливостей відносяться:

- об'єктно-орієнтованість мови. UML ідеально підходить для опису програмного забезпечення, яке буде створено за допомогою об'єктно орієнтованих мов програмування;
- детальність. Мова дозволяє описати програмну систему з усіх сторін, враховуючи усі деталі і нюанси, не впускаючи нічого важливого;
- такі діаграми, при правильній побудові, легкі у сприйнятті і легко розширювані;
- гнучкість. UML гнучка мова, що дозволяє йому додавати нові стандарти, в наслідок чого, мова виводиться за рамки програмної інженерії [22].

В даній роботі для аналітичного супроводження програмного забезпечення використано загальну діаграму прецедентів, що була побудована з використанням стандартів UML [1].

4.2 Розроблення діаграм прецедентів

Основна мета створення будь-якої програмної системи - створення такого програмного продукту, що допомагає користувачеві виконувати свої повсякденні завдання. Для створення таких програм в першу чергу визначаються вимоги, яким повинна задовольняти система. Однак, якщо дати користувачам написати ці вимоги на папері, то часто можна отримати список функцій, за яким важко судити чи майбутня система виконувати своє призначення і чи зможе вона полегшити користувачеві виконання його роботи взагалі.

Для того, щоб більш точно зрозуміти як повинна працювати система, все частіше використовується опис функціональності системи через варіанти використання (Use Case або прецеденти). Варіанти використання це опис послідовності дій, що може здійснювати система у відповідь на зовнішні впливи користувачів або інших програмних систем. Варіанти використання відображають функціональність системи з точки зору отримання результату для користувача, тому вони точніше дозволяють поділяти функції за значимістю одержуваного результату.

Варіанти використання призначені в першу чергу для визначення функціональних вимог до системи і керують усім процесом розробки. Всі основні види діяльності такі як аналіз, проектування, тестування виконуються на основі варіантів використання. Під час аналізу і проектування варіанти використання дозволяють зрозуміти як результати, які хоче отримати користувач впливають на архітектуру системи і як повинні поводитися компоненти системи, для того щоб реалізувати потрібну для користувача функціональність.

В процесі тестування, описані раніше варіанти використання дозволяють оцінити точність реалізації вимог користувачів і дозволяють провести покрокову перевірку цих вимог.

Діаграма варіантів використання складається з акторів, для яких система дає потенційні можливості, крім того ДВВ містить УС, що описують те, що актор хоче отримати від системи.[1]

Окремий варіант використання позначається на діаграмі еліпсом, усередині якого міститься його коротка назва або ім'я у формі дієслова з пояснювальними словами. Мета варіанту використання полягає в тому, щоб визначити закінчений аспект або фрагмент поведінки деякої сутності без розкриття її внутрішньої структури. В якості такої сутності може виступати система або будь-який елемент моделі, що має власну поведінку.

Між акторами і варіантами використання можуть бути різні види взаємодії.

Проста асоціація – являє собою лінію між актором і варіантом використання (без стрілки). Відображає зв'язок актора і варіанти використання. Спрямована асоціація - то ж що і проста асоціація, але показує, що варіант використання викликається актором. Позначається стрілкою.

Генералізація - показує, що нащадок успадкує атрибути і поведінку свого прямого наслідника. Може застосовуватися як для акторів, так для варіантів використання.

Розширення (extend) - показує, що варіант використання розширює базову послідовність дій і вставляє власну послідовність. При цьому на відміну від типу

відносин "включення" розширена послідовність може здійснюватися в залежності від певних умов.

Включення - показує, що варіант використання включається в базову послідовність і виконується завжди.[1]

Створена діаграма прецедентів, що зображена на діаграмі **IA61m.031300.003 Д1** складається з 2х основних підсистем «Сайт для віддаленого управління» та «Об'єкт управління». Розглянемо кожну систему детально.

4.2.1 Розроблення загальної діаграми прецедентів ПКДУ

Розроблена діаграма прецедентів ПКДУ, спрощена модель якої зображена на рис. 1, та рис. 2 відображає загальний перелік акторів, що беруть участь у роботі ПКДУ, прецеденти, доступні кожному з акторів а також взаємозв'язки між ними.

Основна ідея розділу діаграми прецедентів на підсистеми полягає у спрощені сприйняття читачем, взаємозв'язків між підсистемами, а також для наглядного розділу системи.

Ініціатором взаємодії завжди виступає підсистема сайту для віддаленого управління. Актори, використовуючи графічний інтерфейс, надсилають команди управління до підсистеми об'єкт управління, яка їх обробляє та виконує. Результатом виконання команд є надсилання інформації про стан та технічної інформації НРО до сайту для віддаленого управління, де вона відображається у відповідних полях.

4.2.2 Розроблення діаграми прецедентів підсистеми об'єкту управління

Система «Об'єкт управління», спрощену діаграму прецедентів якої зображено на рис. 4.1, відображає можливі варіанти використання, що наявні у НРО, актора об'єкту управління(НРО), та зв'язки між ними.

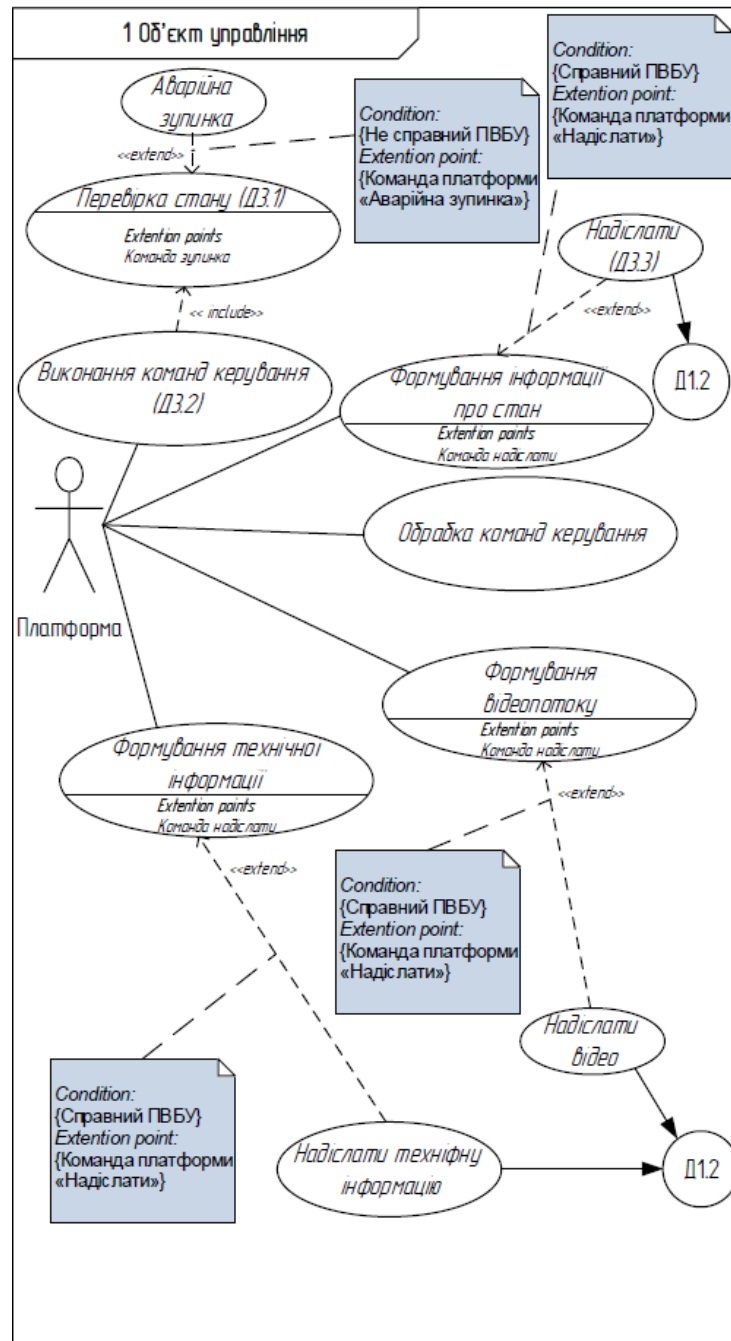


Рисунок 4.1 – Діаграма прецедентів системи «Об'єкт управління»

У наступних таблицях буде розглянуто докладний опис усіх перцедентів.

Прецедент «Виконання команд керування», додатковий опис якого зображено у табл. 4.1, являє собою дії по виконанню отриманих з ВБУ команд керування, за умови справності РО.

Таблиця 4.1 – Опис прецеденту «Виконання команд керування»

<i>Назва (UC)</i>	Виконання команд керування	
<i>Мета або контекст використання</i>	Виконання команд керування, що були надіслані з ВБУ.	
<i>Область видимості та рівень</i>	Для НРО.	
<i>Передумова</i>	Отримано команди керування.	
<i>Умова успішного виконання</i>	Справна платформа, коректна команда управління.	
<i>Умова невиконання</i>	Несправна платформа, помилкова команда управління.	
<i>Первинний, вторинний актор</i>	Платформа.	
<i>Тригер</i>	Відсутні	
<i>Опис (description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Отримання команд керування.
	2	Обробка.
	3	Виконання команд.
<i>Варіанти виконання (sub variant)</i>	Перевірка стану.	

Даний прецедент розкриває основну поведінку роботи РО. Для коретного відпрацювання необхідно провести перевірку системи, що відбувається в автоматичному режимі. При спрацюванні даного прецеденту відбувається надсилання керуючих сигналів до механізмів РО.

Прецедент «Перевірка стану» табл. 4.2, являє собою команду для перевірки стану НРО і виконується перед кожним виконанням команд управління. Цей прецедент має точку розширення «Аварійна зупинка», яка спрацьовує, якщо отримано інформацію про несправність НРО.

Таблиця 4.2 – Опис прецеденту «Перевірка стану»

<i>Назва (UC)</i>	Перевірка стану	
<i>Мета або контекст використання</i>	Проведення огляду технічного стану усіх механізмів РО.	
<i>Область видимості та рівень</i>	На рівні системи.	
<i>Передумова</i>	Запит на виконання команд керування.	
<i>Умова успішного виконання</i>	Наявність живлення на НРО.	
<i>Умова невиконання</i>	Відсутність живлення на НРО.	
<i>Актори</i>	Платформа	
<i>Тригер</i>	Несправність системи.	
<i>Опис (description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запит на перевірку механізмів.
	2	Процес перевірки.
	3	Повернення стану.
<i>Розширення (extention)</i>	Аварійна зупинка	

Шляхом перевірки стану забезпечується впевненість у справності РО. Справною вона вважається, коли усі механізми повернуть інформацію про успішну перевірку. Розширення «Аварійна зупинка», виконує вимкнення подачі живлення на НРО, що виконується у разі несправності хоча б одного механізму РО.

Прецедент «Аварійна зупинка» табл. 4.3 являє собою команду для вимкнення живлення на НРО, при умові отримання несправного стану платформи. Спрацювання відбувається в автоматичному режимі.

Таблиця 4.3 – Опис прецеденту «Аварійна зупинка»

<i>Назва (UC)</i>	Аварійна зупинка	
<i>Мета або контекст використання</i>	Аварійне вимкнення живлення НРО.	
<i>Область видимості та рівень</i>	На рівні системи.	
<i>Передумова</i>	Отримання несправного стану НРО.	
<i>Умова успішного виконання</i>	Справний НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справний НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Платформа.	
<i>Тригер</i>	Несправність платформи	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Отримання несправного стану.
	2	Очікування зупинки усіх механізмів.
	3	Вимкнення живлення на НРО.
<i>Розширення (extention)</i>	-	

Ініціатором виклику даного варіанту використання виступає прецедент перевірка стану. Розглянутий варіант використання важливий тим, що зупиняє роботу усіх механізмів РО при отриманні інформації про несправність хочаб одного механізму. При спрацюванні даного прецеденту відбувається аварійне вимкнення живлення на НРО.

Прецедент «Формування інформації про стан», додатковий опис якого зображено у табл. 4.4, це дії актора, по перевірці стану власного стану та надсилання його у точці розширення «Надіслати».

Таблиця 4.4 – Опис прецеденту «Формування інформації про стан»

<i>Назва (UC)</i>	Формування інформації про стан	
<i>Мета або контекст використання</i>	Збір інформації про стан актора та надсилання її до ВБУ.	
<i>Область видимості та рівень</i>	Для НРО.	
<i>Передумова</i>	Запит на перевірку стану.	
<i>Умова успішного виконання</i>	Увімкнена платформа, справний стан.	
<i>Умова невиконання</i>	Вимкнена платформа.	
<i>Первинний, вторинний актор</i>	Платформа.	
<i>Тригер</i>	Відсутні	
<i>Опис (description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запит на перевірку стану.
	2	Проведено процес перевірки необхідних систем.
	3	Надсилання інформації про стан.
<i>Розширення (extention)</i>	Надіслати.	

В результаті виконання даного прецеденту виконується оформлення отриманої інформації про стан і виклик розширення для надсилання сформованої інформації. При спрацювання розширення відбувається надсилання сформованої інформації про стан НРО. При несправному стані, перед вимкненням подачі живлення на РО, надсилання інформації про стані до ВБУ відбувається у будь-якому випадку.

Прецедент «Надіслати» табл. 4.5, являє собою розширення для прецеденту «Формування інформації про стан», яке надсилає інформацію про стан НРО до ВБУ, за умови виклику команди надсилання.

Таблиця 4.5 – Опис прецеденту «Надіслати»

<i>Назва (UC)</i>	Надіслати	
<i>Мета або контекст використання</i>	Відправлення інформації про стан справності НРО до ВБУ.	
<i>Область видимості та рівень</i>	На рівні системи.	
<i>Передумова</i>	Виконання формування інформації про стан НРО.	
<i>Умова успішного виконання</i>	Справний НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справний НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Платформа.	
<i>Тригер</i>	Виклик команди надіслати	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Отримання інформації про стан НРО.
	2	Формування пакету для надсилання.
	2	Надсилання інформації до ВБУ.
<i>Розширення (extention)</i>	-	

У результаті виконання даного прецеденту відбувається надсилання інформації про стан НРО до ВБУ. Надсилання відбувається у будь-якому випадку, навіть при несправності механізмів НРО. Єдиним виключенням може бути відсутність інтернет зв'язку, в такому випадку інформацію не буде доставлено.

Прецедент «Формування відеопотоку», додатковий опис якого зображено у табл. 4.6, це дії актора, по отриманні відео з власної камери, його обробка та надсилання у точці розширення «Надіслати».

Таблиця 4.6 – Опис прецеденту «Формування відеопотоку»

<i>Назва (UC)</i>	Формування відеопотоку	
<i>Мета або контекст використання</i>	Отримання інформації з камери, форматування та її надсилання до ВБУ.	
<i>Область видимості та рівень</i>	Для НРО.	
<i>Передумова</i>	Запит на отримання відео з камери.	
<i>Умова успішного виконання</i>	Увімкнена платформа, справний стан.	
<i>Умова невиконання</i>	Вимкнена платформа, несправний стан.	
<i>Первинний, вторинний актор</i>	Платформа.	
<i>Тригер</i>	Відсутні	
<i>Опис (description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запит на отримання відео.
	2	Зчитування відео з камери, обробка.
	3	Надсилання відео.
<i>Розширення (extention)</i>	Надіслати відео.	

Оскільки для формування відео у високій роздільній здатності потрібен високопродуктивний процесор та відеоадаптер, то у нашому випадку буде використовуватись нижча роздільна здатність, але достатня для зрозумілого відображення необхідних елементів. При спрацюванні розширення відбувається надсилання сформованого відеопотоку з камери НРО до ВБУ.

Прецедент «Надіслати відео» табл. 4.7, являє собою розширення для прецеденту «Формування відеопотоку», яке надсилання відео з камери НРО до ВБУ за умови виклику команди надсилання.

Таблиця 4.7 – Опис прецеденту «Надіслати відео»

<i>Назва (UC)</i>	Надіслати відео	
<i>Мета або контекст використання</i>	Відправлення сформованого відео з камери НРО до ВБУ.	
<i>Область видимості та рівень</i>	На рівні системи.	
<i>Передумова</i>	Виконання формування відео з камери НРО.	
<i>Умова успішного виконання</i>	Справний НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справний НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Платформа.	
<i>Тригер</i>	Виклик команди надіслати відео	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Отримання сформованого відео з камери НРО.
	2	Надсилання відео до ВБУ.
<i>Розширення (extention)</i>	-	

В результаті спрацювання даного прецеденту відбувається надсилання відео з камери НРО до ВБУ. Оскільки передача ведеться по протоколу UDP, то при низькій швидкості інтернету можливі затримки, не повна передача відео та зниження його якості. Для уникнення цього – необхідно забезпечити канал з достатньою швидкістю передачі даних.

Прецедент «Формування технічної інформації», додатковий опис якого зображено у табл. 4.8, це дії актора, по отриманні технічних даних від механізмів, їх обробка та надсилання у точці розширення «Надіслати технічну інформацію».

Таблиця 4.8 – Опис прецеденту «Формування технічної інформації»

<i>Назва (UC)</i>	Формування технічної інформації	
<i>Мета або контекст використання</i>	Отримання інформації від механізмів, форматування та її надсилання до ВБУ.	
<i>Область видимості та рівень</i>	Для НРО.	
<i>Передумова</i>	Запит на отримання технічної інформації.	
<i>Умова успішного виконання</i>	Увімкнена платформа, справний стан.	
<i>Умова невиконання</i>	Вимкнена платформа, несправний стан.	
<i>Первинний, вторинний актор</i>	Платформа.	
<i>Тригер</i>	Відсутні	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запит на отримання технічної інформації.
	2	Зчитування інформації з механізмів.
	3	Надсилання технічної інформації.
<i>Розширення (extention)</i>	Надіслати технічну інформацію.	

Технічна інформація формується на основі даних, що зчитуються з необхідних механізмів. Зчитана інформація оформлюється у узгодженій формі. При спрацювання розширення відбувається надсилання сформованої технічної інформації НРО до ВБУ.

Прецедент «Надіслати технічну інформацію» табл. 4.9, являє собою розширення для прецеденту «Формування технічної інформації», яке надсилає технічну інформацію НРО до ВБУ за умови виклику команди надсилання.

Таблиця 4.9 – Опис прецеденту «Надіслати технічну інформацію»

<i>Назва (UC)</i>	Надіслати технічну інформацію	
<i>Мета або контекст використання</i>	Відправлення інформації про стан справності НРО до ВБУ.	
<i>Область видимості та рівень</i>	На рівні системи.	
<i>Передумова</i>	Виконання формування інформації про технічний стан НРО.	
<i>Умова успішного виконання</i>	Справний НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справний НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Платформа.	
<i>Тригер</i>	Виклик команди надіслати інформацію про технічний стан	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Отримання сформованого технічного стану НРО.
	2	Надсилання інформації про технічний стан до ВБУ.
<i>Розширення (extention)</i>	-	

Отримані технічні дані про НРО поміщуються у пакет для надсилання та надсилаються. Результатом роботи даного прецеденту є надсилання сформованої технічної інформації НРО до ВБУ.

Прецедент «Обробка команд керування», додатковий опис якого зображено у табл. 4.10, являє собою дії по обробці отриманих з ВБУ команд керування та їх перетворенню.

Таблиця 4.10 – Опис прецеденту «Обробка команд керування»

<i>Назва (UC)</i>	Обробка команд керування	
<i>Мета або контекст використання</i>	Обробка команд керування, що були надіслані з ВБУ та формування сигналів управління.	
<i>Область видимості та рівень</i>	Для НРО.	
<i>Передумова</i>	Отримано команди керування.	
<i>Умова успішного виконання</i>	Справна платформа, коректна команда управління.	
<i>Умова невиконання</i>	Несправна платформа, помилкова команда управління.	
<i>Первинний, вторинний актор</i>	Платформа.	
<i>Тригер</i>	Відсутні	
<i>Опис (description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Отримання команд керування.
	2	Формування керуючих сигналів.
	3	Надсилання керуючих сигналів на виконання.
<i>Варіанти виконання (sub variant)</i>	—	

Інформація, що являє собою команди керування, передається з ВБУ в узгодженому форматі тому формування керуючих сигналів відбувається по заготовленій процедурі.

4.2.2.1 Програмна компонента рухомого об'єкту

ПК рухомого об'єкту має складатися з програми мікроконтролера, що керує рухомих об'єктом. У будь-якому випадку можна створити систему програмних інтерфейсів і протоколів для узагальненого управління РО у системі ПКДУ.

На даному етапі ПК рухомого об'єкту змодельована у вигляді двох файлів, один з яких накопичує отриману інформацію для налагодження ПКДУ у цілому, інший моделює відповіді на запити користувача до РО.

У прототипі ПКДУ побудована модель періодичного опитування стану РО для виявлення несправностей. При виявленні несправності спрацьовує аварійна зупинка.

У перспективі для забезпечення управління у реальному часі можна вибрати одну із наступних технологій і протоколів для обміну даними:

- HTTP;
- виключно TCP;
- WebSockets;
- FTP.

У створеному прототипі буде використано HTTP, який у повній мірі задовольняє швидкість передачі даних і дозволяє виконувати оновлення з потрібною частотою.

Створений макет буде містити лише програму, що буде імітувати роботу реального НРО. Хоч дане рішення менш наглядне, проте його використання дозволяє у повній мірі розкрити усі аспекти керування за допомогою віддаленого сайту управління, докладний опис якого буде розміщений у наступних розділах.

Для відображення результатів роботи НРО, окрім надсилання технічного стану до ВБУ, виконується запис інформації до файлу.

4.2.3 Розроблення діаграма прецедентів веб-додатку

Підсистема сайту для віддаленого управління, спрощену діаграму прецедентів якої зображено на рис. 4.2, відображає можливі варіанти використання, що наявні у користувачів ВБУ, можливих користувачів ВБУ та взаємозв'язки між ними.

Основними дійовими особами (акторами) даної діаграми рис. 4.2 є «Адміністратор», «Оператор», «Глядач». Актор «Глядач» є права тільки на огляд інформації про робота і огляд загальної інформації. «Оператор» має усі можливості глядача, а крім того він може керувати рухом платформи. «Адміністратор» має усі можливості як глядача так і оператора, а також він може реєструвати нових користувачів.

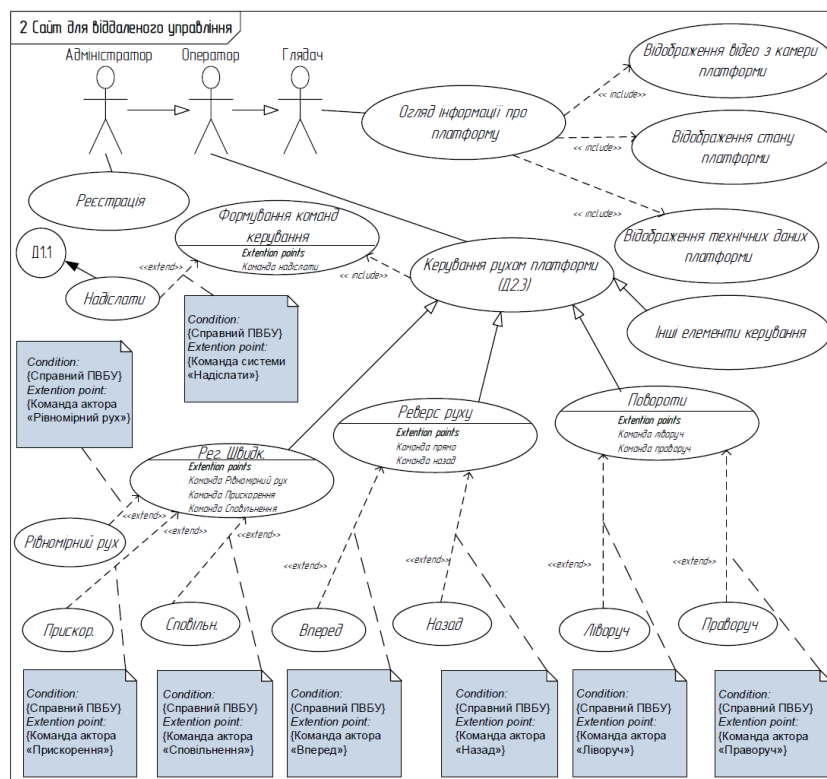


Рисунок 4.2 – Діаграма прецедентів системи «Сайт для віддаленого управління»

Опишемо основні прецеденти(UC) даної діаграми за допомогою таблиць опису прецедентів (UCDT). Такі таблиці, що також використовувались при описі прецедентів підсистеми об'єкту управління, докладно описують основну інформацію про кожен з прецедентів підсистеми і системи в загальному.

Прецедент «Огляд інформації про платформу», додатковий опис якого зображено у табл. 4.11, відображає змогу актора переглядати основну інформацію про стан, переглядати відео з камери та переглядати технічні дані НРО.

Таблиця 4.11 – Опис прецеденту «Огляд інформації про платформу»

<i>Назва (UC)</i>	Огляд інформації про платформу	
<i>Мета або контекст використання</i>	Відображення основної інформації про платформу, такої як відео з камери платформи, стану платформи, технічних даних платформи.	
<i>Область видимості та рівень</i>	Для усіх акторів ДВУ адміністратора оператора глядача.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна платформа, наявні права доступу.	
<i>Умова невиконання</i>	Несправна платформа, відсутні права доступу.	
<i>Первинний, вторинний актор</i>	Оператор, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутні.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Отримання необхідної інформації з НРО.
<i>Варіанти виконання (sub variant)</i>	1. Відображення відео з камери платформи. 2. Відображення стану платформи. 3. Відображення технічних даних платформи.	

Даний прецедент спрямований на відображення відео з камери НРО, технічної інформації та інформації про стан, що періодично оновлюється, або при надсиланні команд керування.

Прецедент «Відображення відео з камери платформи», додатковий опис якого зображено у табл. 4.12, відображає змогу актора переглядати відео з камери НРО у режимі реального часу.

Таблиця 4.12 – Опис прецеденту «Відображення відео з камери платформи»

<i>Назва (UC)</i>	Відображення відео з камери платформи	
<i>Мета або контекст використання</i>	Відображення відео з камери.	
<i>Область видимості та рівень</i>	Для усіх акторів ДВУ адміністратора оператора глядача.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна платформа, наявні права доступу.	
<i>Умова невиконання</i>	Несправна платформа, відсутні права доступу.	
<i>Первинний, вторинний актор</i>	Оператор, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутні.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control»
	2	Отримання необхідної інформації з НРО
	3	Відображення відео.
<i>Варіанти виконання (sub variant)</i>	-	

Відео, що є результатом виконання даного прецеденту, відображається у спеціальній формі на сторінці «Control» сайту для віддаленого управління. Передача відбувається у режимі реального часу, з відповідним рівнем роздільної здатності.

Прецедент «Відображення стану платформи», додатковий опис якого зображено у табл. 4.13, відображає зможу актора переглядати інформацію про стан НРО, який передається з НРО через певний період часу.

Таблиця 4.13 – Опис прецеденту «Відображення стану платформи»

<i>Назва (UC)</i>	Відображення стану платформи	
<i>Мета або контекст використання</i>	Відображення інформації про справність НРО, для інформування актора.	
<i>Область видимості та рівень</i>	Для усіх акторів ДВУ адміністратора оператора глядача.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна платформа, наявні права доступу.	
<i>Умова невиконання</i>	Несправна платформа, відсутні права доступу.	
<i>Первинний, вторинний актор</i>	Оператор, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутні.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control»
	2	Отримання необхідної інформації з НРО
	3	Відображення інформації на сторінці
<i>Варіанти виконання (sub variant)</i>	—	

Для успішного використання даного прецеденту, користувач повинен бути авторизованим та мати необхідні права доступу. При розробці сайту буде забезпечено яскраве відображення стану, для того, щоб при несправності користувач відразу звернув увагу на відповідну інформацію і прийняв необхідні дії. Інформація про несправність надсилається з НРО до вимкнення живлення.

Прецедент «Відображення технічних даних платформи», додатковий опис якого зображено у табл. 4.14, відображає зможу актора переглядати технічні дані НРО, до яких відносяться швидкість, напрям руху та координати.

Таблиця 4.14 – Опис прецеденту «Відображення технічних даних платформи»

<i>Назва (UC)</i>	Відображення технічних даних платформи	
<i>Мета або контекст використання</i>	Відображення технічних даних платформи, швидкості, напрямку руху та координати.	
<i>Область видимості та рівень</i>	Для усіх акторів ДВУ адміністратора оператора глядача.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна платформа, наявні права доступу.	
<i>Умова невиконання</i>	Несправна платформа, відсутні права доступу.	
<i>Первинний, вторинний актор</i>	Оператор, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутні	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control»
	2	Отримання необхідної інформації з НРО
	3	Відображення технічних даних НРО
<i>Варіанти виконання (sub variant)</i>	—	

Технічні дані платформи приймаються у певній обговорені формі, зрозумілій для НРО і ВБУ. Для перегляду даних користувач повинен мати відповідний рівень прав. Надсилання інформації про технічний стан відбувається після виконання кожної команди управління або через певний період часу, при відсутності таких команд.

Прецедент «Огляд головної сторінки», додаткова інформація про який описана на табл. 4.15, спрямований на відображення останніх новин, зображення головної інформації ВБУ.

Таблиця 4.15 – Опис прецеденту «Огляд головної сторінки»

<i>Назва (UC)</i>	Огляд головної сторінки	
<i>Мета або контекст використання</i>	Відображення Основної інформації та останніх новин ВБУ.	
<i>Область видимості та рівень</i>	Для усіх зареєстрованих відвідувачів сайту.	
<i>Передумова</i>	Відкриття головної сторінки сайту ВБУ.	
<i>Умова успішного виконання</i>	Наявний доступ до інтернету, запущений сайт, справна база даних.	
<i>Умова невиконання</i>	Проблеми з доступом до інтернету, незапущений сайт, несправна база даних.	
<i>Первинний, вторинний актор</i>	Користувач, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутній	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запуск сайту.
	2	Авторизація.
	3	Відкриття головної сторінки сайту .
<i>Варіанти виконання (sub variant)</i>	—	

Умовою потрапляння на головну сторінку сайту є проходження авторизації. Основна інформація, що зображена на головній сторінці, доступна для усіх рівнів доступу і являє собою останні новини, огляд НРО та інформацію про них.

Прецедент «Огляд Гайду», додаткова інформація про який описана на табл. 4.16, спрямований на відображення інструкції по керуванню НРО та інформацію про данні відображені на сторінці «Control».

Таблиця 4.16 – Опис прецеденту «Огляд Гайду»

<i>Назва (UC)</i>	Огляд Гайду	
<i>Мета або контекст використання</i>	Відображення інструкції по керуванню НРО та інформацію про данні НРО.	
<i>Область видимості та рівень</i>	Для усіх зареєстрованих відвідувачів сайту.	
<i>Передумова</i>	Відкриття сторінки «Guide» ВБУ.	
<i>Умова успішного виконання</i>	Наявний доступ до інтернету, запущений сайт, справна база даних.	
<i>Умова невиконання</i>	Проблеми з доступом до інтернету, незапущений сайт, несправна база даних.	
<i>Первинний, вторинний актор</i>	Користувач, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутній	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запуск сайту.
	2	Авторизація.
	3	Відкриття сторінки сайту «Guide» .
<i>Варіанти виконання (sub variant)</i>	—	

Огляд гайду доступний для усіх авторизованих користувачів. Гайд представляє собою опис можливої функціональності, що доступна користувачам з рівнем доступу, достатнім для управління НРО, та технічну інформацію про стан платформи.

Прецедент «Огляд додаткової інформації», додаткова інформація про який описана на табл. 4.17, спрямований на відображення інформації про авторів ВБУ, містить контакти та актуальні адреса електронної пошти ВБУ.

Таблиця 4.17 – Опис прецеденту «Огляд додаткової інформації»

<i>Назва (UC)</i>	Огляд додаткової інформації	
<i>Мета або контекст використання</i>	Відображення інформації про творців ВБУ та їх контактні дані.	
<i>Область видимості та рівень</i>	Для усіх зареєстрованих відвідувачів сайту.	
<i>Передумова</i>	Відкриття сторінки «About» сайту ВБУ.	
<i>Умова успішного виконання</i>	Наявний доступ до інтернету, запущений сайт, справна база даних.	
<i>Умова невиконання</i>	Проблеми з доступом до інтернету, незапущений сайт, несправна база даних.	
<i>Первинний, вторинний актор</i>	Користувач, Глядач, Адміністратор.	
<i>Тригер</i>	Відсутній	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Запуск сайту.
	2	Авторизація.
	3	Відкриття сторінки сайту «About».
<i>Варіанти виконання (sub variant)</i>	—	

Для використання даного прецеденту необхідно авторизуватися, та перейти на сторінку «About» ВБУ. Інформація на цій сторінці являє собою дані про розробника, місце його роботи, контактні дані, телефон, електронна адреса та дату їх актуальності.

Одним з головних прецедентів системи «Сайт для віддаленого управління» є «Керування рухом платформи» табл. 4.18. Даний прецедент є узагальнюючим для усіх прецедентів управління платформою.

Таблиця 4.18 – Опис прецеденту «Керування рухом платформи»

<i>Назва (UC)</i>	Керування рухом платформи	
<i>Мета або контекст використання</i>	Загальна змога керування НРО, узагальнення усіх можливих варіантів управління.	
<i>Область видимості та рівень</i>	Для авторизованих користувачів з достатнім рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справа НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Первинний, вторинний актор</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопок управління.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки управління.
	3	Перевірка стану НРО.
	4	Формування команди управління та надсилання до НРО.
<i>Варіанти виконання (sub variant)</i>	1. Перевірка стану платформи. 2. Формування команд керування.	

Розглянутий вище прецедент є узагальнюючим, оскільки усі перцеденти, що представляють управління, пов'язані з ним типом зв'язку генералізація.

Прецедент «Реверс руху» табл. 4.19, один з варіантів управління НРО. Цей прецедент являє собою загальний варіант управління реверсом руху, з точками розширення «Вперед» та «Назад».

Таблиця 4.19 – Опис прецеденту «Реверс руху»

<i>Назва (UC)</i>	Реверс руху	
<i>Мета або контекст використання</i>	Відправлення команди для реверсу руху до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки для руху вперед або назад.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопок реверсу руху.
	3	Перевірка стану НРО.
	4	Формування команди реверсу руху та надсилення до НРО.
<i>Розширення (extention)</i>	1. Вперед. 2. Назад.	

Реверс руху визначає напрям у якому буде рухатись НРО. Виділено два основних напрями, це напрям Вперед та напрям Назад. Оскільки РО є наземними, то необхідність у русі Вгору та Вниз відпадає.

Прецедент «Вперед» табл. 4.20, один з варіантів управління НРО. Цей прецедент являє собою команду для руху НРО вперед.

Таблиця 4.20 – Опис прецеденту «Вперед»

Назва (UC)	Вперед	
Мета або контекст використання	Відправлення команди для руху вперед до НРО.	
Область видимості та рівень	Для користувачів з відповідним рівнем доступу.	
Передумова	Відкриття сторінки «Control».	
Умова успішного виконання	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
Умова невиконання	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
Актори	Оператор, Адміністратор.	
Тригер	Натискання кнопки руху вперед.	
Опис(description)	Кроки	Дія
	1	Відкриття сторінки «Control».
	2	Натискання кнопки руху вперед.
	3	Перевірка стану НРО.
	4	Формування команди руху вперед та надсилання до НРО.
Розширення (extention)	-	

При спрацюванні даного прецеденту відбувається надсилання команди про рух вперед до НРО. Команда передається у вигляді коду, що узгоджений між ВБУ та НРО. Якщо НРО перед цим рухався у проилежному напрямку, швидкість скидається до 0 і починається набір до відповідної швидкості вперед.

Прецедент «Назад» табл. 4.21, один з варіантів управління НРО. Цей прецедент являє собою команду для руху НРО назад.

Таблиця 4.21 – Опис прецеденту «Назад»

<i>Назва (UC)</i>	Назад	
<i>Мета або контекст використання</i>	Відправлення команди для руху назад до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки руху назад.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки руху назад.
	3	Перевірка стану НРО.
	4	Формування команди руху назад та надсилання до НРО.
<i>Розширення (extention)</i>	-	

Результатом спрацювання даного прецеденту є надсилання команди про рух назад до НРО. Команда передається у вигляді коду, що узгоджений між ВБУ та НРО. Якщо НРО перед цим рухався у проилежному напрямку, швидкість скидається до 0 і починається набір до відповідної швидкості назад.

Прецедент «Повороти» табл. 4.22, один з варіантів управління НРО. Цей прецедент являє собою загальний варіант управління поворотами, з точками розширення «Ліворуч» та «Праворуч».

Таблиця 4.22 – Опис прецеденту «Повороти»

Назва (UC)	Повороти	
Мета або контекст використання	Відправлення команди для повороту до НРО.	
Область видимості та рівень	Для користувачів з відповідним рівнем доступу.	
Передумова	Відкриття сторінки «Control».	
Умова успішного виконання	Справа НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
Умова невиконання	Не справа НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
Актори	Оператор, Адміністратор.	
Тригер	Натискання кнопки для повороту праворуч або ліворуч.	
Опис(description)	Кроки	Дія
	1	Відкриття сторінки «Control».
	2	Натискання кнопок поворотів.
	3	Перевірка стану НРО.
	4	Формування команди повороту та надсилання до НРО.
Розширення (extention)	1. Ліворуч. 2. Праворуч.	

Прецедент, що описаний вище, викликає процес формування команд управління поворотами та надси до НРО. Усі команди узгоджені між НРО та ВБУ.

Прецедент «Ліворуч» табл. 4.23, один з варіантів управління НРО. Цей прецедент являє собою команду для повороту НРО ліворуч.

Таблиця 4.23 – Опис прецеденту «Ліворуч»

<i>Назва (UC)</i>	Ліворуч	
<i>Мета або контекст використання</i>	Відправлення команди для повороту ліворуч до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки ліворуч.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки ліворуч.
	3	Перевірка стану НРО.
	4	Формування команди ліворуч та надсилення до НРО.
<i>Розширення (extention)</i>	-	

При спрацюванні даного прецеденту відбувається надсилення команди про поворот ліворуч до НРО. Команда являє собою числовий код, узгоджений з НРО, який відповідає певній дії, яку виконає НРО. У відповідь на виконання буде надіслано необхідно інформацію з НРО.

Прецедент «Праворуч» табл. 4.24, один з варіантів управління НРО. Цей прецедент являє собою команду для повороту НРО праворуч.

Таблиця 4.24 – Опис прецеденту «Праворуч»

<i>Назва (UC)</i>	Праворуч	
<i>Мета або контекст використання</i>	Відправлення команди для повороту праворуч до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки праворуч.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки праворуч.
	3	Перевірка стану НРО.
	4	Формування команди повороту праворуч та надсилання до НРО.
<i>Розширення (extention)</i>	-	

Результатом спрацювання даного прецеденту є надсилання команди про поворот праворуч до НРО. Команда являє собою числовий код, узгоджений з НРО, який відповідає певній дії, яку виконає НРО. У відповідь на виконання буде надіслано необхідно інформацію з НРО.

Складовою керування НРО є прецедент «Регулювання швидкості», додаткову інформацію про який описано на табл. 4.25. Даний прецедент є узагальнюючим для прецедентів керування швидкістю руху.

Таблиця 4.25 – Опис прецеденту «Регулювання швидкості»

Назва (UC)	Регулювання швидкості	
Мета або контекст використання	Узагальнення для прецедентів керування швидкістю.	
Область видимості та рівень	Для авторизованих користувачів з достатнім рівнем доступу.	
Передумова	Відкриття сторінки «Control».	
Умова успішного виконання	Справа НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
Умова невиконання	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
Актори	Оператор, Адміністратор.	
Тригер	Натискання кнопок управління.	
Опис(description)	Кроки	Дія
	1	Відкриття сторінки «Control».
	2	Натискання кнопки керування швидкістю.
	3	Перевірка стану НРО.
	4	Формування команди управління та надсилання до НРО.
Розширення (extention)	1. Рівномірний рух. 2. Прискорення. 3. Сповільнення	

Регулювання швидкості відбувається поступово, відповідно до надісланих сигналів.

Прецедент «Рівномірний рух» табл. 4.26, один з варіантів управління НРО.
Цей прецедент являє собою команду для рівномірного руху НРО.

Таблиця 4.26 – Опис прецеденту «Рівномірний рух»

<i>Назва (UC)</i>	Рівномірний рух	
<i>Мета або контекст використання</i>	Відправлення команди для рівномірного руху до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справа НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справа НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки рівномірного руху	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки рівномірного руху.
	3	Перевірка стану НРО.
	4	Формування команди рівномірного руху та надсилання до НРО.
<i>Розширення (extention)</i>	-	

При спрацюванні даного прецеденту відбувається надсилання команди про рівномірний рух до НРО. Рівномірний рух відображає відсутність необхідності НРО змінювати швидкість руху, та підтримувати її на тому рівні, на якому вона була при отриманні цієї команди.

Прецедент «Прискорення» табл. 4.27, один з варіантів управління НРО. Цей прецедент являє собою команду для прискорення руху НРО.

Таблиця 4.27 – Опис прецеденту «Прискорення»

Назва (UC)	Прискорення	
Мета або контекст використання	Відправлення команди для рівномірного руху до НРО.	
Область видимості та рівень	Для користувачів з відповідним рівнем доступу.	
Передумова	Відкриття сторінки «Control».	
Умова успішного виконання	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
Умова невиконання	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
Актори	Оператор, Адміністратор.	
Тригер	Натискання кнопки прискорення швидкості руху	
Опис(description)	Кроки	Дія
	1	Відкриття сторінки «Control».
	2	Натискання кнопки прискорення швидкості руху.
	3	Перевірка стану НРО.
	4	Формування команди прискорення швидкості руху та надсилання до НРО.
Розширення (extention)	-	

Результатом спрацювання даного прецеденту є надсилання команди про прискорення до НРО. Такий сигнал інформує НРО про необхідність збільшувати попередню швидкість до тих змін, поки не прийде інша команда управління швидкістю.

Прецедент «Сповільнення» табл. 4.28, один з варіантів управління НРО. Цей прецедент являє собою команду для сповільнення руху НРО.

Таблиця 4.28 – Опис прецеденту «Сповільнення»

Назва (UC)	Сповільнення	
Мета або контекст використання	Відправлення команди для рівномірного руху до НРО.	
Область видимості та рівень	Для користувачів з відповідним рівнем доступу.	
Передумова	Відкриття сторінки «Control».	
Умова успішного виконання	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
Умова невиконання	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
Актори	Оператор, Адміністратор.	
Тригер	Натискання кнопки сповільнення швидкості руху	
Опис(description)	Кроки	Дія
	1	Відкриття сторінки «Control».
	2	Натискання кнопки сповільнення швидкості руху.
	3	Перевірка стану НРО.
	4	Формування команди сповільнення швидкості руху та надсилання до НРО.
Розширення (extention)	-	

Після виклику цього прецеденту відбувається надсилання команди про сповільнення швидкості руху до НРО. Сповільнення, для НРО, символізує зменшення поточної швидкості до тих пір, поки не прийде інша команда керування швидкістю.

Прецедент «Перевірка стану» табл. 4.29, один зі складових контролю НРО. Цей прецедент являє собою команду для запиту стану НРО і виконується перед кожним надсиланням команд управління. Цей прецедент має точку розширення «Аварійна зупинка», яка спрацьовує, якщо отримано інформацію про несправність платформи.

Таблиця 4.29 – Опис прецеденту «Перевірка стану»

<i>Назва (UC)</i>	Перевірка стану	
<i>Мета або контекст використання</i>	Відправлення запиту про стан до НРО.	
<i>Область видимості та рівень</i>	На рівні системи.	
<i>Передумова</i>	Натискання кнопок управління.	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету.	
<i>Умова невиконання</i>	Відсутній доступ до інтернету.	
<i>Актори</i>	Оператор, Адміністратор, Глядач.	
<i>Тригер</i>	Несправність системи.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Перевірка стану НРО.
	3	Відображення стану.
<i>Розширення (extention)</i>	Аварійна зупинка	

Перевірка стану це команда для НРО що символізує собою ініціацію процесу перевірки механізмів. В разі, якщо прийде відповідь про несправність системи, спрацює розширення «Аварійна зупинка», що надсилає сигнал про зупинку до НРО, при виникненні несправностей.

Прецедент «Аварійна зупинка» табл. 4.30, один з варіантів управління НРО. Цей прецедент являє собою команду для зупинки НРО, при умові отримання несправного стану платформи.

Таблиця 4.30 – Опис прецеденту «Аварійна зупинка»

<i>Назва (UC)</i>	Аварійна зупинка	
<i>Мета або контекст використання</i>	Відправлення команди для аварійної зупинки до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Несправність платформи	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Перевірка стану НРО.
	3	Формування команди аварійної зупинки та надсилання до НРО.
<i>Розширення (extention)</i>	-	

Аварійна зупинка спрацьовує в разі отримання від НРО інформації про несправність. Даний прецедент надсилає команду до НРО про вимкнення її вимкнення, після якої НРО повинний зупинити усі механізми і вимкнути подачу живлення.

Прецедент «Зупинка» табл. 4.31, один з варіантів управління НРО. Цей прецедент являє собою команду для зупинки НРО.

Таблиця 4.31 – Опис прецеденту «Зупинка»

<i>Назва (UC)</i>	Зупинка	
<i>Мета або контекст використання</i>	Відправлення команди для зупинки до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки зупинки	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки зупинки.
	3	Перевірка стану НРО.
	4	Формування команди зупинки та надсилання до НРО.
<i>Розширення (extention)</i>	Зупинка.	

У даному прецеденті, при спрацюванні точки розширення «Зупинка», на НРО надсилається відповідна команда про зупинку. Зупинка символізує зменшення швидкості НРО до 0. В залежності від конструкції НРО, можливі повороти на місці.

Прецедент «Реєстрація» табл. 4.32, являє собою реєстрацію нових користувачів. Цей прецедент доступний лише для «Адміністратора».

Таблиця 4.32 – Опис прецеденту «Реєстрація»

<i>Назва (UC)</i>	Реєстрація	
<i>Мета або контекст використання</i>	Додавання нових користувачів сайту.	
<i>Область видимості та рівень</i>	Для Адміністратора.	
<i>Передумова</i>	Доступ до бази даних.	
<i>Умова успішного виконання</i>	Достатній рівень доступу користувача, справна база даних.	
<i>Умова невиконання</i>	Недостатній рівень доступу користувача, несправна база даних.	
<i>Актори</i>	Адміністратор	
<i>Тригер</i>	Відсутній	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття системи управління базою даних
	2	Внесення необхідних даних.
	3	Збереження введених даних
<i>Розширення (extention)</i>	-	

Реєстрація символізує процес додавання нового користувача до бази даних. Реєструвати нових користувачів має право авторизований користувач з рівнем доступу «Адміністратор». Для отримання даних для авторзації необхідно подати запит до адміністратора.

Прецедент «Формування команд керування», докладний опис якого описано на табл. 4.33, являє собою команду на формування відповідних команд для надсилання їх до НРО.

Таблиця 4.33 – Опис прецеденту «Формування команд керування»

Назва (UC)	Формування команд керування	
Мета або контекст використання	У відповідь на натискання кнопок керування відбувається формування відповідних команд, для надсилання до НРО.	
Область видимості та рівень	Для користувачів з відповідним рівнем доступу.	
Передумова	Відкриття сторінки «Control».	
Умова успішного виконання	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
Умова невиконання	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
Актори	Оператор, Адміністратор.	
Тригер	Натискання кнопки керування.	
Опис(description)	Кроки	Дія
	1	Відкриття сторінки «Control».
	2	Натискання кнопки керування.
	3	Перевірка стану НРО.
	4	Формування відповідної команди.
Розширення (extention)	Надіслати.	

Формування команд керування це прецедент, під час дії якого відбувається створення інформаційного блоку, а основі інформації отриманої в результаті натискання кнопок інтерфейсу управління. При спрацюванні точки розширення «Надіслати», на НРО надсилається відповідна команда керування.

Прецедент «Надіслати» табл. 4.34, один з варіантів управління НРО. Цей прецедент являє собою розширення для прецеденту «Формування команд керування», яке надсилає сформовану команду керування до НРО.

Таблиця 4.34 – Опис прецеденту «Надіслати»

<i>Назва (UC)</i>	Надіслати	
<i>Мета або контекст використання</i>	Відправлення команди до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справа НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Несправність платформи	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Перевірка стану НРО.
	3	Формування команди управління.
	4	Надсилання команд до НРО.
<i>Розширення (extention)</i>	-	

В результаті виконання цього прецеденту формуються пакети даних та відправляються цільовому НРО. З боку НРО відбувається обробка цих даних та формування сигналів управління.

Складовою керування НРО є прецедент «Управління підсвіткою», додаткову інформацію про який описано на табл. 4.35. Даний прецедент є узагальнюючим для прецедентів керування підсвіткою.

Таблиця 4.35 – Опис прецеденту «Управління підсвіткою»

<i>Назва (UC)</i>	Управління підсвіткою	
<i>Мета або контекст використання</i>	Узагальнення для прецедентів керування підсвіткою.	
<i>Область видимості та рівень</i>	Для авторизованих користувачів з достатнім рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопок управління підсвіткою.	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки керування підсвіткою.
	3	Перевірка стану НРО.
	4	Формування команди управління та надсилання до НРО.
<i>Розширення (extention)</i>	1. Вимкнути підсвітку. 2. Ближнє світло. 3. Дальнє світло.	

Результатом цього прецедета є надсилання необхідних команд керування підсвіткою, що передбачені на НРО.

Прецедент «Вимкнути підсвітку» табл. 4.36, один з варіантів управління підсвіткою НРО. Цей прецедент являє собою команду для вимкнення підсвітки на НРО.

Таблиця 4.36 – Опис прецеденту «Вимкнути підсвітку»

<i>Назва (UC)</i>	Вимкнути підсвітку	
<i>Мета або контекст використання</i>	Відправлення команди для вимкнення підсвітки до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справа НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки вимкнення підсвітки	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки вимкнення підсвітки.
	3	Перевірка стану НРО.
	4	Формування команди вимкнення підсвітки та надсилання до НРО.
<i>Розширення (extention)</i>	—	

При спрацюванні даного прецеденту відбувається надсилання команди про вимкнення підсвітки до НРО. Така команда имволізує вимкнення підсвітки загалом на НРО.

Прецедент «Ближнє світло» табл. 4.37, один з варіантів управління НРО.
Цей прецедент являє собою команду для ввімкнення ближнього світла на НРО.

Таблиця 4.37 – Опис прецеденту «Ближнє світло»

<i>Назва (UC)</i>	Ближнє світло	
<i>Мета або контекст використання</i>	Відправлення команди для ввімкнення ближнього світла до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки ввімкнення ближнього світла	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки ввімкнення ближнього світла.
	3	Перевірка стану НРО.
	4	Формування команди ввімкнення ближнього світла та надсилання до НРО.
<i>Розширення (extention)</i>	-	

Результатом спрацювання даного прецеденту є надсилання команди про ввімкнення ближнього світла до НРО. Умова виконання такого прецедента зі сторони НРО є наявність необхідної апаратної частини.

Прецедент «Дальнє світло» табл. 4.38, один з варіантів управління НРО. Цей прецедент являє собою команду для сповільнення руху НРО.

Таблиця 4.38 – Опис прецеденту «Дальнє світло»

<i>Назва (UC)</i>	Дальнє світло	
<i>Мета або контекст використання</i>	Відправлення команди для ввімкнення дальнього світла до НРО.	
<i>Область видимості та рівень</i>	Для користувачів з відповідним рівнем доступу.	
<i>Передумова</i>	Відкриття сторінки «Control».	
<i>Умова успішного виконання</i>	Справна НРО, наявний доступ до інтернету, достатній рівень доступу користувача.	
<i>Умова невиконання</i>	Не справна НРО, відсутній доступ до інтернету, недостатній рівень доступу користувача.	
<i>Актори</i>	Оператор, Адміністратор.	
<i>Тригер</i>	Натискання кнопки ввімкнення дальнього світла	
<i>Опис(description)</i>	<i>Кроки</i>	<i>Дія</i>
	1	Відкриття сторінки «Control».
	2	Натискання кнопки ввімкнення дальнього світла.
	3	Перевірка стану НРО.
	4	Формування команди ввімкнення дальнього світла та надсилення до НРО.
<i>Розширення (extention)</i>	-	

При спрацюванні даного прецеденту відбувається надсилення команди про ввімкнення дальнього світла до НРО.

4.2.3.1 Розроблення Web додатку для віддаленого управління

На сьогоднішній день для розробників Web додатків є великий вибір як технологій для розробки так і мов програмування, на яких можна реалізовувати Web додатки. Для прикладу можна навести Perl, PHP, ASP.NET, ASP.NET Core, JSP, Coldfusion тощо.

Основною технологією, що була використана нами для створення Web додатку прототипу ПКДУ, є технологія ASP.NET Core MVC [23].

ASP.NET – це платформа, що надає усі необхідні служби для створення серверних web-додатків. Вона створена на основі платформи .NET Framework, тому усі функції цієї платформи доступні для додатків ASP.NET.

Використання ASP не вимагає специфічних браузерів. Всі ASP-скрипти запускаються і виконуються на веб-сервері, причому браузер отримує тільки оброблені та сформовані HTML-файли. Microsoft Internet Information Server, починаючи з версії 3.0, підтримує ASP додатки.

Функціонування ASP має наступну послідовність. Клієнт надсилає запит на ASP-сторінку до веб-серверу. Сервер приймає запит і починає його обробляти. ASP-код, в свою чергу, може містити звернення до різних джерел даних, здійснювати обробку отриманих даних і додавати вміст генерується сторінки. В результаті формується "звичайна" HTML-сторінка (вже не містить ASP-коду), яка і відправляється назад клієнту.

Зовні ASP функціонує так само як і CGI. При надсиланні даних від форми, він кодує вхідні дані, а сценарій CGI декодує їх, функціонально обробляє і повертає вихідні дані браузеру.

Аналогічним чином передаються параметри (формат рядка запиту) і здійснюється виведення результатів. Однак продуктивність ASP є набагато вищою, тому, що при кожному запиті не відбувається окремого завантаження ASP-інтерпретатора.

Крім підвищення продуктивності ASP також вирішує проблему формування динамічних web-сторінок. Раніше при використанні CGI доводилося вбудовувати

текст оформлення web-сторінки в програмний код CGI-сценарію, що ускладнювало подальшу зміну дизайну, або змушувало розробників створювати свої власні системи шаблонів. Використання ASP дозволяє розробнику одночасно працювати над програмним кодом і над оформленням Web-сторінки[24].

Для змоги масштабування додатку було використано патерн MVC 5, концепція якого полягає у розділі додатку на три компоненти.

Контролер являє собою клас, що забезпечує зв'язок між користувачем та системою, відображенням та базою даних. Він отримує введені користувачем дані та обробляє їх. В залежності від результатів обробки відправляє користувачу певний результат, наприклад у вигляді відображення.

Відображення – це візуальна частина або інтерфейс користувача, що являє собою HTML сторінку, за допомогою якої, користувачеві надсилається уся інформація.

Модель являє собою клас, у якому описується логіка усіх даних, що використовується у web-додатку.

При такому підході модель являється незалежним компонентом, при чому будь які зміни контролера або відображення не впливають на модель. Контролер і відображення являються відносно незалежними компонентами, що дає змогу змінювати їх в не залежності один від одного.

Завдяки цьому реалізується концепція розділу відповідальності, в наслідок чого простіше організовувати роботу над окремими компонентами. Крім того, з використанням такого розділу, стає простіше тестувати такий додаток.

На основі вище сказаного, можна сформулювати основні переваги використання патерну MVC, які полягають у простоті масштабування проекту, жорсткому контролі над HTML і HTTP, легкості тестування, досконалості системи маршрутизації [25].

ASP.NET Core являється кросплатформним, високопродуктивним середовищем з відкритим кодом для створення сучасних хмарних додатків. Воно дозволяє виконувати наступні завдання:

- створювати web-додатки та служби, додатки інтернету речей та серверні частини мобільних додатків;
- використовувати різні засоби для розробки для Windows, macOS та Linux;
- розміщувати додатки на хмарі або локальному середовищі;
- працювати з .NET Core або .NET Framework. [23]

Дана технологія є останньою розробкою корпорації Microsoft, основною перевагою якої є кросплатформеність. Вона дозволяє розробляти Web додатки, використовуючи мову програмування C# та платформу .NET. Обрана технологія дозволяє розміщати Web-додатки на серверах POSIX сумісних ОС.

Додатково, для забезпечення необхідної функціональності, було використано бібліотеку JQuery мови Javascript, для використання функцій Ajax.

AJAX (Asynchronous JavaScript and XML) - це спосіб надсилання даних на сервер і отримання відповіді без перезавантаження сторінки, що називається асинхронними запитами. AJAX не є мовою програмування, а являє собою концепцію обміну даними з сервером.

На певному етапі роботи скрипта JavaScript на клієнтській стороні викликається функція, яка відправляє запит на сервер і вказує який скрипт запустити на стороні сервера (Asp.Net Core). На сервері запускається скрипт, за результатами роботи клієнту приходить відповідь і скрипт на стороні клієнта продовжує роботу.

AJAX істотно полегшує роботу Web-додатку, коли потрібно відправити дані форми. Якщо це робити звичайним способом, то після відправки форми необхідно буде прийняти відповідь сервера і завантажити новий документ у вікні. При використанні AJAX новий документ не завантажується - а це означає, що кількість даних може бути значно знижено. Це особливо критично для мобільних пристроїв, які використовують мобільний зв'язок і обмежені в швидкості передачі даних.

Для збереження даних було використано базу даних MySql, яка являється базою даних з відкритим вихідним кодом, окрім декількох останніх функціональних модулів.

MySQL - це найпоширеніша повноцінна серверна СУБД. MySQL має досить широку функціональність і використовується з різними сайтами та Web-додатками.

MySQL являється клієнт-сервер системою, що містить багато-SQL-серверів, в наслідок чого забезпечується підтримка різних обчислювальних машин баз даних, а також кілька різних клієнтських програм і бібліотек, засоби адміністрування і широкий спектр програмних інтерфейсів. Також наявна багатопоточна версія MySQL, яку можна підключити до необхідного додатку і отримати компактний, більш швидкий і легкий в управлінні продукт. Доступна також велика кількість додаткового ПЗ для MySQL.

До основних переваг MySQL над іншими системами управління базами даних можна віднести масштабованість, легку зв'язність та швидкість функціонування. Також варто відмітити, що вона легка у переміщенні та має достатньо високий рівень безпеки [26].

4.2.3.2 Графічний інтерфейс користувача

Як було сказано раніше, користувач, маючи доступ до мережі через звичайний персональний комп'ютер, може здійснювати управління РО. Для цього у прототипі ПКДУ передбачений графічний інтерфейс користувача рис. 4.3.

Графічний інтерфейс користувача, як програмна компонента ПКДУ, основні кнопки управління якої відповідають за:

- швидкість (прискорення, сповільнення і рівномірний рух);
- реверс руху(вперед і назад);
- повороти(праворуч, ліворуч);
- зупинка.

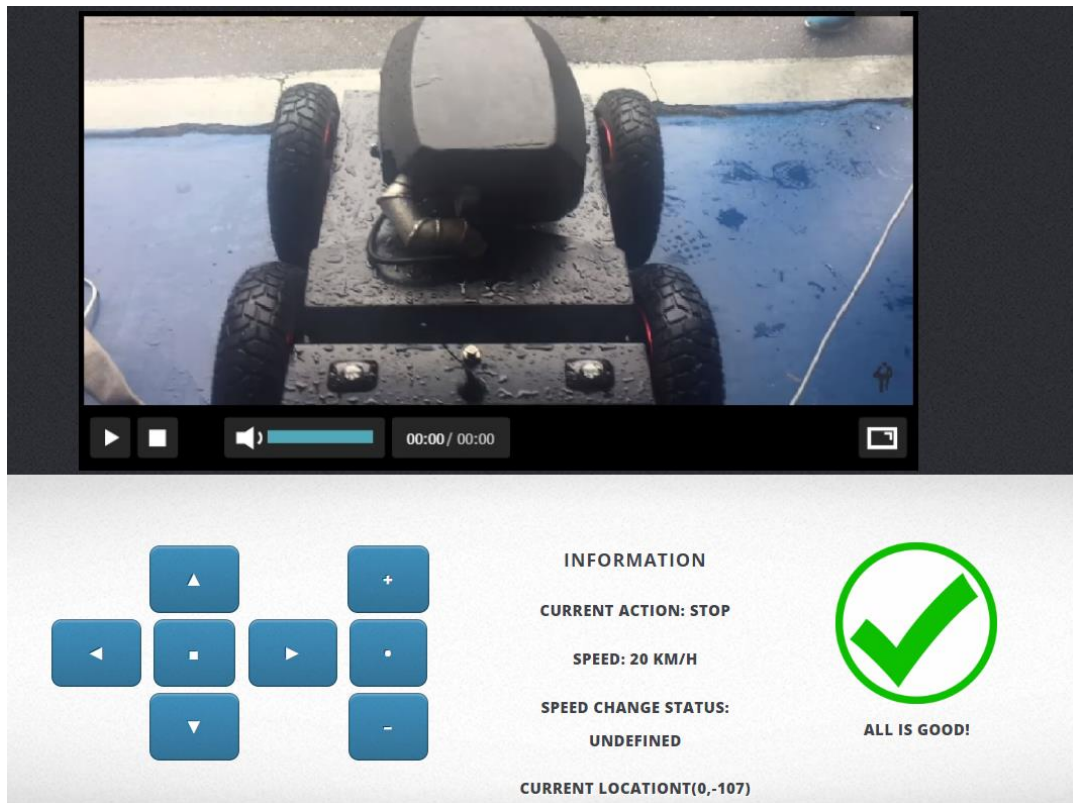


Рисунок 4.3 – Графічний інтерфейс «Сайту для віддаленого управління»

Крім графічного інтерфейсу, наявне управління за допомогою клавіатури, що реалізовано з технологіями JavaScript(JQuery). Оператор самостійно вирішує який спосіб доцільно обрати для управління рухом.

Як можна побачити з рис. 4.3 реалізовано інтерфейс відображення поточної інформації з камери спостереження. Крім того, програмна компонента ПКДУ дає можливість відображення технічної інформації про системи РО, що включає в себе:

- поточну дію, що виконує платформа;
- швидкість платформи;
- стан зміни швидкості платформи;
- поточну локацію(координати).

Для забезпечення високої швидкості оновлення інформації і її передачі використовується технологія Ajax запитів, що реалізована у JQuery. Вона дозволяє асинхронно, без перенавантаження інтерфейсу, викликати Web сервіси і методи, що забезпечують обмін інформації без оновлення усієї сторінки.

Програмні компоненти РО побудовані таким чином, що отримують сигнали з Web-додатку, обробляють їх перетворюючи у сигнали управління, а також формують інформацію про стан РО, після чого надсилають його до Web додатку.

Крім того, для більшого розуміння відображуваної інформації оператором, наявна сторінка «Guide» рис. 4.4.

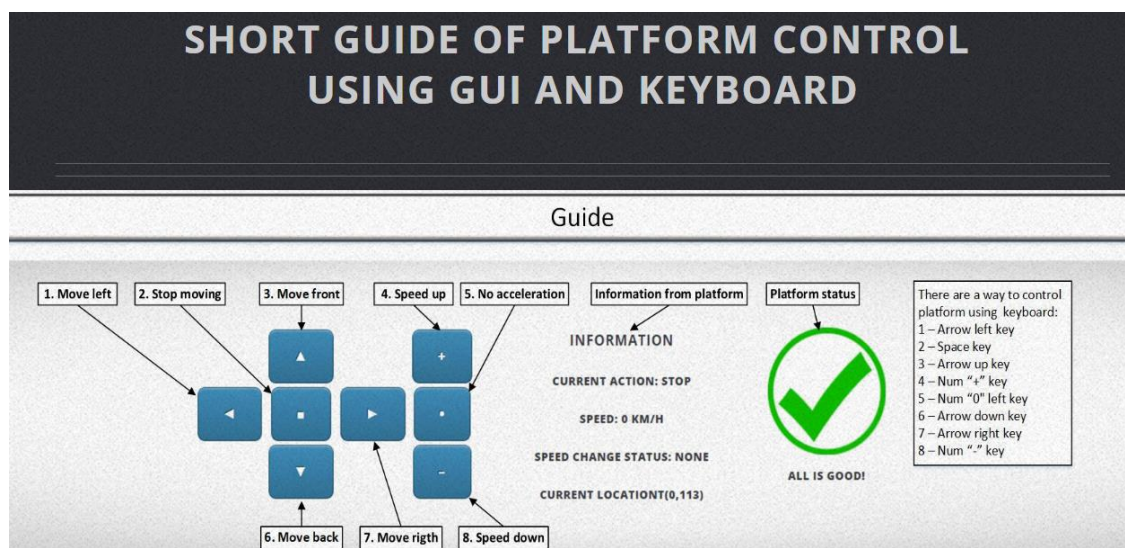


Рисунок 4.4 – Сторінка «Guide» «Сайту для віддаленого управління»

Дана сторінка спрямована на інформування авторизованих користувачів сайту, про спрямованість кнопок управління, значення виведеної інформації та додаткова інформація про особливості управління клавіатурою.

Головною сторінкою сайту являється сторінка «Home», зображена на рис. 4.5. Розробка сайту для віддаленого управління велася з використанням правил ергономіки, для зручності і простоти сприйняття графічного інтерфейсу.

До основних цілей ергономіки слід віднести підвищення ефективності системи, правильний розподіл функції між людиною і технічними засобами, задіяти необхідні функції організму (мислення, увагу, пам'ять, свідомість). В даний час розрізняють мікроергономіку і макроергономіку. Мікроергономіка орієнтована на вивчення і проектування інтерфейсів "людина - інший компонент робочої системи".

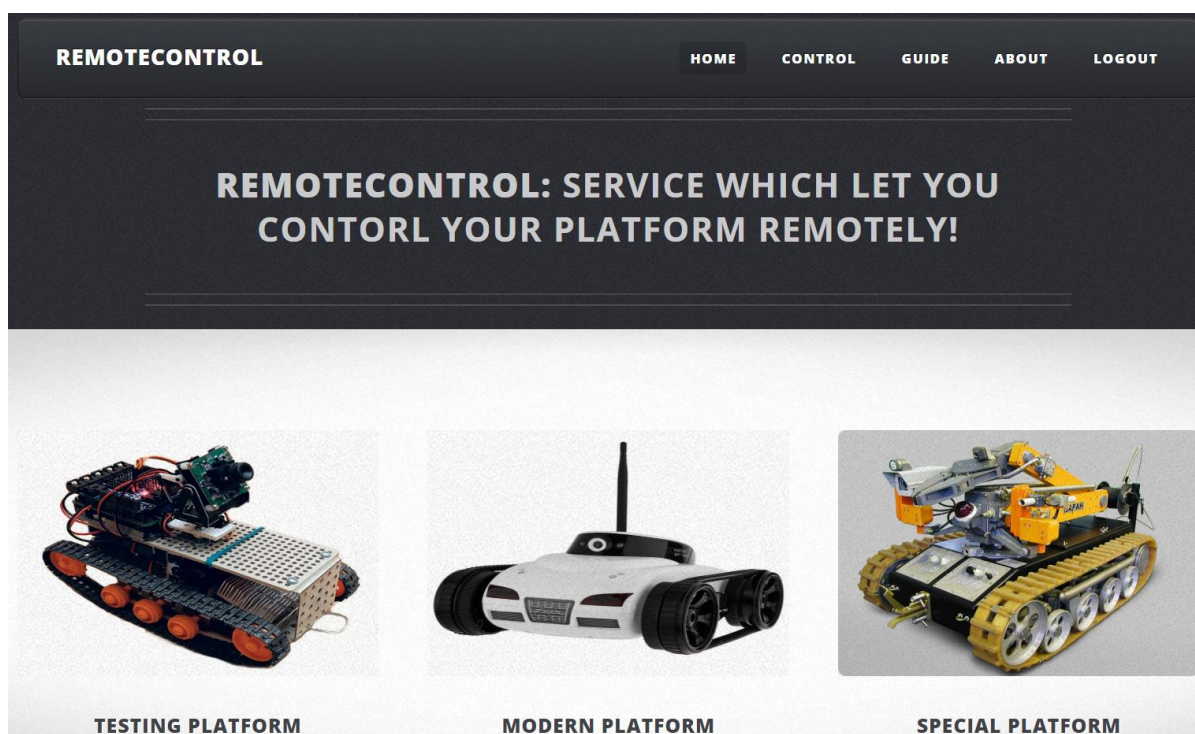


Рисунок 4.5 – Сторінка «Home» «Сайту для віддаленого управління»

На головні сторінці сайту відображаються останні новини, актуальні РО, тощо.

Відомі різні види інтерфейсів, наприклад, інтерфейс людина - робота", "людина-машина", "людина - програмне забезпечення", "людина - середовище". Ергономіка вивчає проблеми оптимального розподілу та узгодження функцій між людиною і машиною, проектує процес діяльності людини, обґрунтовує оптимальні вимоги до засобів і умов діяльності і розробляє методи їх обліку при створенні і експлуатації техніки, керованої і обслуговується людиною. Ергономічні вимоги передбачаються відповідними нормативними документами, найчастіше стандартами [27].

5 ФОРМАЛІЗАЦІЯ ПОВЕДІНКИ ПКДУ ЗА ДОПОМОГОЮ UML ДІАГРАМ

Для моделювання поведінки на логічному рівні в мові UML можуть використовуватися відразу кілька канонічних діаграм: станів, діяльності, послідовності і кооперації, кожна з яких фіксує увагу на окремому аспекті функціонування системи. На відміну від інших діаграм діаграма станів описує процес зміни станів тільки одного класу, а точніше - одного примірника певного класу, тобто моделює всі можливі зміни в стані конкретного об'єкта. При цьому зміна стану об'єкта може бути викликано зовнішніми впливами з боку інших об'єктів або ззовні. Саме для опису реакції об'єкта на подібні зовнішні впливи і використовуються діаграми станів.

Головне призначення цієї діаграми - описати можливі послідовності станів і переходів, які в сукупності характеризують поведінку елемента моделі протягом його життєвого циклу. Діаграма станів представляє динамічну поведінку сутностей, на основі специфікації їх реакції на сприйняття деяких конкретних подій. Системи, які реагують на зовнішні впливи від інших систем або від користувачів, називають реактивними. Якщо такі дії ініціюються в довільні випадкові моменти часу, то говорять про асинхронному поведінці моделі.

Хоча діаграми станів найчастіше використовуються для опису поведінки окремих екземплярів класів (об'єктів), але вони також можуть бути застосовані для специфікації функціональності інших компонентів моделей, таких як варіанти використання, актори, підсистеми, операції і методи. Діаграма станів по суті є графом спеціального виду, який представляє певний автомат.

Автомат (state machine) в мові UML є певний формалізм для моделювання поведінки елементів моделі і системи в цілому. В метамоделі UML автомат є пакетом, в якому визначено безліч понять, необхідних для подання поведінки модельованої суті виді дискретного простору з кінцевим числом станів і переходів. З іншого боку, автомат описує поведінку окремого об'єкта в формі послідовності станів, які охоплюють всі етапи його життєвого циклу, починаючи

від створення об'єкта і закінчуючи його знищенням. Кожна діаграма станів представляє певний автомат [1].

5.1 Математична формалізація технічної системи у вигляді FSM

FSM (finite state machine) або кінцевий автомат це метод моделювання систем, вихід яких залежить від історії їхніх входів, а не тільки від останніх вхідних даних. У порівнянні з функціональними системами, в яких вихідний сигнал визначається вхідним сигналом, FSM має властивість, що визначається її історією. FSM можуть бути використані для моделювання широкого спектру систем, у тому числі:

- інтерфейси користувача, із вводом з клавіатури, натисканням кнопок мишки, тощо;
- стани космічного корабля, включаючи стан клапанів(відкриті або закриті), рівень мастила, кисню, тощо;

Розглянемо примітивну машину станів.

Ми можемо вказати перетворювач (процес, який приймає на вхід послідовність значень, які служать в якості вхідних даних для машини станів, на виході повертає вихідний набір для кожного вводу) як кінцевий автомат, вказавши:

- набір станів S ;
- набір вхідних даних I , який називають вхідним словником;
- набір вихідних даних O , який називають вихідним словником;
- функцію переходу $n(i, s)$, що поверта наступний стан;
- функцію виходу $o(i, s)$, що повертає вихідне значення;
- вихідне значення s .

Найпростіший вид кінцевого автомату - це чиста функція: якщо автомат не має станів, а функція виводу є суто функцією входу, наприклад, $o = i + 1$, то ми

маємо безпосередньо функціональний зв'язок між входами та виходами в один і той же час [28].

Кінцеві автомати дозволяють побудувати моделі системи паралельної обробки, однак, щоб змінити кількість паралельних процесів у такій моделі, потрібно внести істотні зміни у саму модель. Крім того, спроба розробки складної моделі на кінцевому автоматі призведе до швидкого зростання кількості станів автомата, що в підсумку зробить розробку такої моделі недоцільною. Цю проблему можна вирішити, якщо використовувати недетермінований автомат.

Розглянемо приклади примітивних FSM.

Першим прикладом буде лічильник підйомів та спусків. Його основним функціоналом буде обрахунок кількості підйомів і спусків.

Відповідно до необхідних компонентів, для створення кінцевих автоматів, вкажемо початкові значення:

S = ціле число;

$I = \{u, d\}$;

O = ціле число;

$n(s, i) = s+1$, якщо $i = u$ або $s-1$, якщо $i=d$;

$o(s, i) = n(s, i)$;

Результати роботи такого кінцевого автомату для вхідної послідовності u, u, u, d, d, u , відображено на табл. 5.1.

Таблиця 5.1 – Результати роботи кінцевого автомату лічильника

t	0	1	2	3	4	5	6
i	u	u	u	d	d	u	
s	0	1	2	3	2	1	2
o	1	2	3	2	1	2	

Наступним прикладом буде кінцевий автомат для затримки, основним завданням якого є на вихід видавати вхідне значення, але із затримкою в 1 крок.

Тому k -й елемент вхідної послідовності буде $k+1$ елементом вихідної послідовності. Формальний вигляд такого атомату:

$S = \text{будь що};$

$I = \text{будь що};$

$O = \text{будь що};$

$n(s, i) = i;$

$o(s, i) = s;$

$s \text{ вихідний} = 0;$

Якщо на вхід такого кінцевого автомату подати послідовність i_0, i_1, i_2, \dots , то він сформує вихідну послідовність вигляду $0, i_0, i_1, i_2, \dots$. Першим на виході є 0 тому, що FSM повинний мати можливість виводити вихідний результат, навіть коли попередніх вхідних даних не було, тому у якості першого виходу є вихідне значення.

Результати роботи такого кінцевого автомату для вхідної послідовності 3, 1, 2, 5, 9 відображено на табл. 5.2.

Таблиця 5.2 – Результати роботи кінцевого автомату для затримки

t	0	1	2	3	4	5
i	3	1	2	5	9	
s	0	3	1	2	5	9
o	0	3	1	2	5	

Ще одним прикладом кінцевого автомату може бути акумулятор, основним завданням якого є сумувати усі вхідні значення. Опишемо формальний вигляд такого автомату:

$S = \text{числа};$

$I = \text{числа};$

$O = \text{числа};$

$n(s, i) = s+i;$

$o(s, i) = n(s+i);$

s вихідний = 0;

Результати роботи такого кінцевого атомату для вхідної послідовності 100, -3, 4, -123, 10 відображено на табл. 5.3.

Таблиця 5.3 – Результати роботи кінцевого атомату для затримки

t	0	1	2	3	4	5
i	100	-3	4	-123	10	
s	0	100	97	101	-22	-12
o	100	97	101	-22	-12	

Наступним прикладом кінцевого атомату, буде кінцевий атомат оснований на функції знаходження середнього значення, який зберігає попереднє вхідне значення у якості стану. Формальний вигляд такого атомату:

S = числа;

I = числа;

O = числа;

$n(s, i) = i$;

$o(s, i) = (s+i)/2$;

s вихідний = 0;

Результати роботи такого кінцевого атомату для вхідної послідовності 100, -3, 4, -123, 10 відображено на табл. 5.4.

Таблиця 5.4 – Результати роботи кінцевого атомату для затримки

t	0	1	2	3	4	5
i	100	-3	4	-123	10	
s	0	100	-3	4	-123	10
o	50	48,5	0,5	-59,5	-56,5	

Отже, кінцеві атомати є загальною формою формалізації цифрових систем. Використання однієї системи для визначення примітивних атоматів та

комбінацій дає змогу простіше проводити формалізацію програмного забезпечення.

5.2 UML Діаграма станів «Сайту для віддаленого управління»

В процесі створення прототипу було створено дві основні діаграми станів, діаграма станів «Сайту для віддаленого управління» [IA61м.031300.003 Д2](#) та діаграма станів «Рухомого об'єкту» [IA61м.031300.003 Д3](#).

Діаграма станів «Сайту для віддаленого управління» рис. 5.1 відображає стани, у яких перебуває ВБУ протягом своєї роботи.

Перший загальний стан, у який переходить система після відкриття сайту - це стан «Не авторизовано». У цьому стані, у відвідувачів сайту наявно змога переглядати лише сторінку авторизації, вводити дані, необхідні для авторизації і запускати процес перевірки введених даних для завершення процесу авторизації. У разі введення хибних даних, не буде знайдено користувача з відповідними даними, після чого буде відображено відповідну помилку про відсутність користувача з такими даними.

Якщо данні введені вірні, тобто знайдений відповідний користувач у базі даних, то буде відображено головну сторінку сайту і виконано перехід у стан «Авторизовано».

Загальний стан «Авторизовано» починається з переходу системи у стан «Очікування запиту на перехід між сторінками сайту».

При переході користувача на сторінку «Control» відбувається перехід системи у загальний підстан «Керування платформою», у якому першим кроком відбувається запит на отримання інформації про стан платформи. Якщо інформацію про стан платформи прийнято – відбувається перехід у стан «Огляд інформації та очікування на натискання кнопок управління».

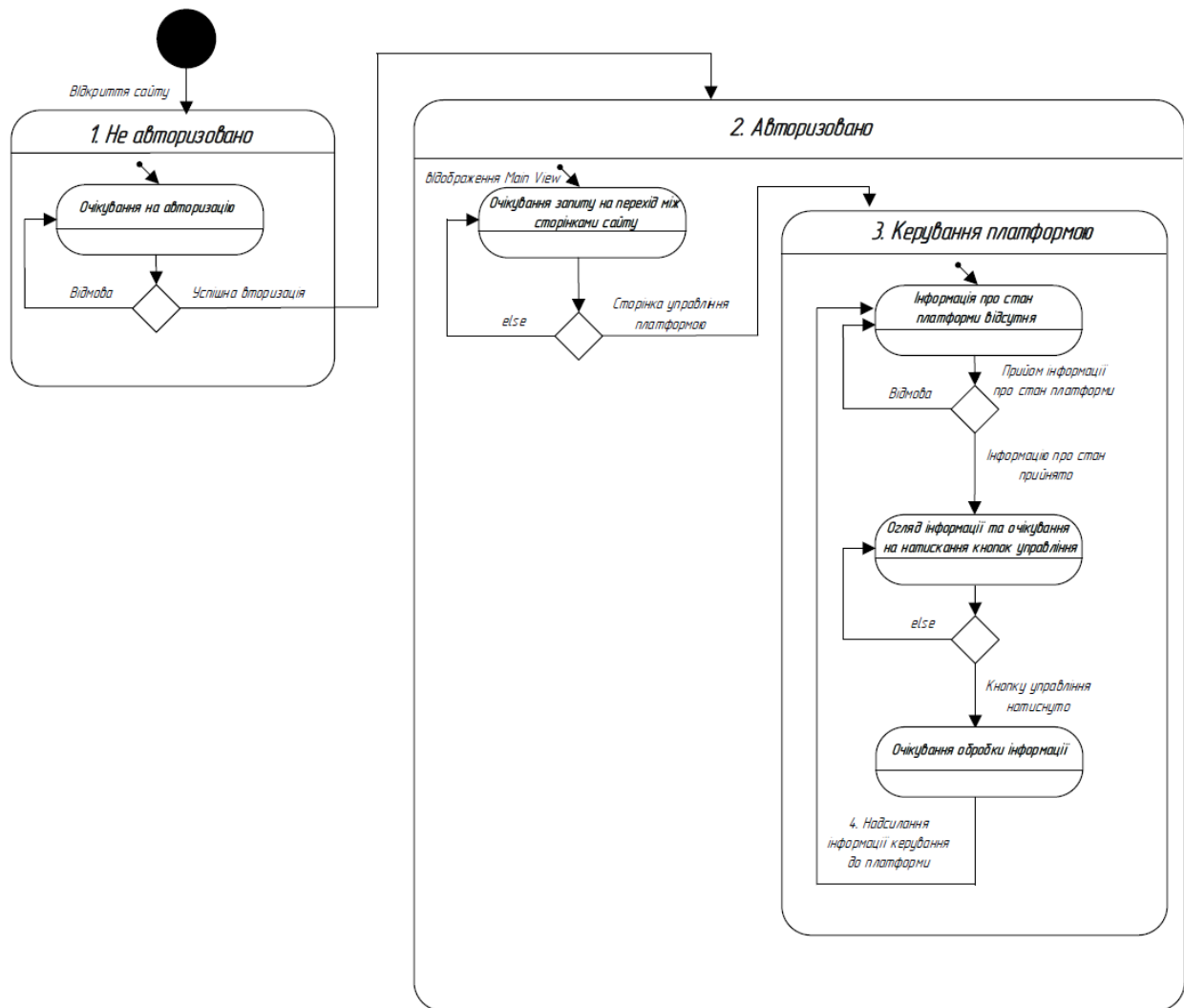


Рисунок 5.1 – Діаграма станів «Сайту для віддаленого управління»

У разі, якщо у користувача достатньо прав для управління і він натискає кнопку для управління – відбувається перехід у стан «Очікування обробки інформації», у якому система формує команди управління та надсилає їх до НРО, після чого знову проводиться перехід у стан «Інформація про стан платформи відсутня».

Тобто основним станом платформи, являється узагальнюючий підстан «Керування платформою», у якому відбуваються основні дії ВБУ.

5.3 UML Діаграма станів «Рухомого об'єкту»

Діаграма станів «Рухомого об'єкту» рис. 5.2 відображає стани, у яких перебуває НРО протягом своєї роботи.

Перший загальний стан, у який переходить система після увімкнення живлення – це стан «Невизначений стан». У цьому стані система ще не знає чи усі компоненти рухомого об'єкту справні. Даний стан містить підстан «Перевірка стану платформи», у якому відбувається перевірка усіх систем, для визначення справності. У разі відмови – система переходить у стан «Непрацездатний стан» у якому відбувається надсилання інформації про непрацездатність рухомого об'єкта і вимикається живлення.

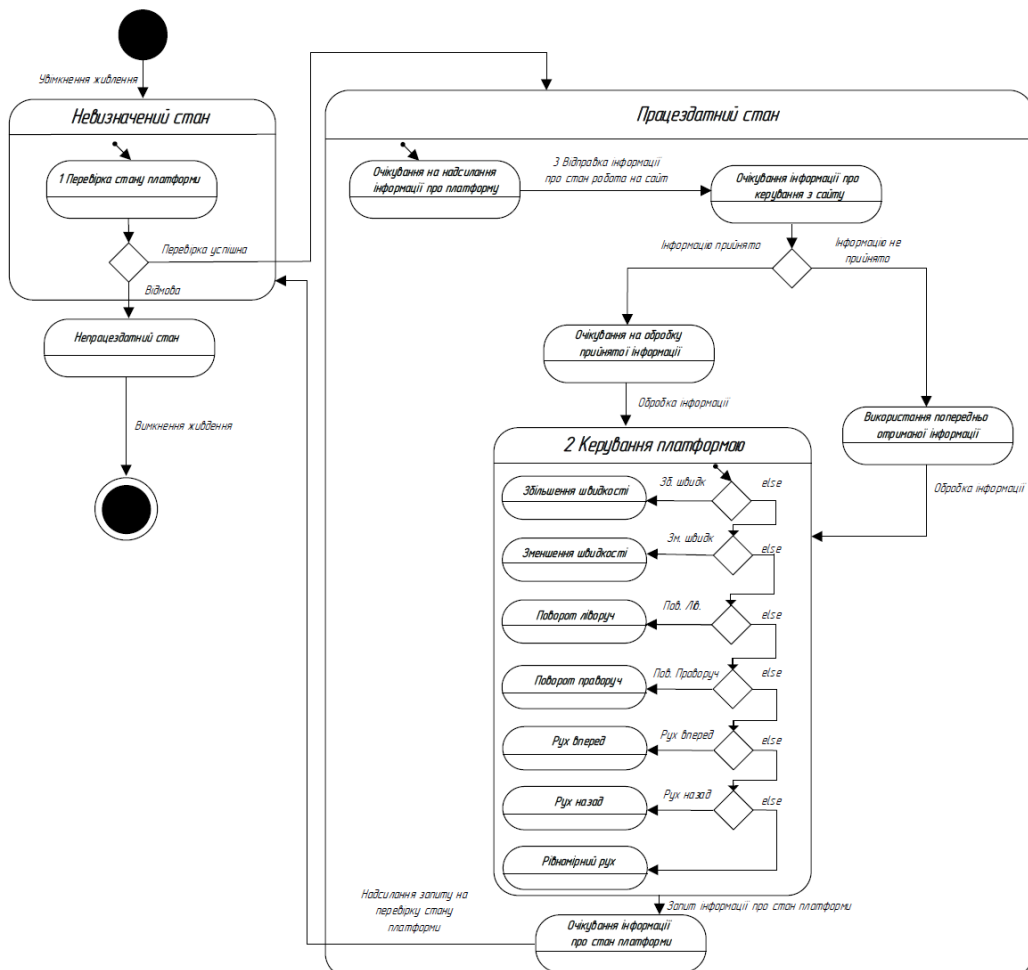


Рисунок 5.2 – Діаграма станів «Рухомого об'єкту»

При успішній перевірці стану – система переходить у загальний стан «Працездатний стан» у першому підстані («Надсилання інформації про стан платформи») відбувається надсилання інформації про стан НРО до ВБУ.

Після надсилання інформації, відбувається перехід у стан «Очікування інформації про керування з сайту», якщо на протязі певного фіксованого часу не надходить нова інформація про керування – НРО виконує попередні команди, якщо приходить нова інформація, то у стані «Очікування на обробку інформації» відбувається обробка інформації і формування необхідних сигналів управління платформою. Після чого повторюється перевірка інформації про стан НРО.

5.4 Розроблення діаграм послідовностей

Діаграма послідовностей є найбільш поширеним видом діаграми взаємодії, яка фокусується на обміні повідомленнями між декількома лініями життєзабезпечення.

На діаграмі послідовності зображаються тільки ті об'єкти, які безпосередньо беруть участь у взаємодії. Ключовим моментом для діаграм послідовності є динаміка взаємодії об'єктів в часі.

Діаграма послідовності є однією з різновиду діаграм взаємодії та призначена для моделювання взаємодії об'єктів Системи в часі, а також обміну повідомленнями між ними.

Одним з основних принципів ООП є спосіб інформаційного обміну між елементами Системи, що виражається у відправці і отриманні повідомлень один від одного. Таким чином, основні поняття діаграми послідовності пов'язані з поняттям Об'єкт і Повідомлення.

На діаграмі послідовності, кожен учасник представлений разом зі своєю лінією життя (lifeline), це вертикальна лінія під об'єктом, вертикально впорядковує повідомлення на сторінці. Важливо: всі повідомлення на діаграмі слід читати зверху вниз. Кожна лінія життя має смугу активності (прямокутники), яка показує інтервал активності кожного учасника при взаємодії.

Учасники діаграми іменуються наступним чином: ім'я: Клас, де і ім'я, і клас є не обов'язковими, але якщо використовується клас, то присутність двокрапки обов'язково.

Під час побудови діаграм послідовностей було створено дві діаграми. Перша діаграма **IA61м.031300.003 Д4** відображає загальну послідовність дій при здійсненні управління. Друга **IA61м.031300.003 Д5** відображає докладний варіант дій при здійсненні управління з розглядом варіантів управління.

5.4.1 Розроблення діаграми послідовності (загальної)

Загальна діаграма послідовностей рис. 5.3 містить 4 основні об'єкти, між якими відбувається взаємодія, а саме:

- Користувач
- WebBrowser
- Controller
- Platform

Користувач являє собою будь яку людину, що заходить на сайт, WebBrowser – це програма, яку використовує користувач для входу на сайт. Controller це головна програма управління сайтом, яка виконує усі функції(обробки інформації, прийому та відповіді на запити користувачів, взаємодії з платформою). Platform – це наша НРО.

Опишемо дану діаграму докладніше. Для відкриття сайту, користувач надсилає запит на отримання головної сторінки сайту через браузер. У відповідь на такий запит, Controller повертає HTML сторінку авторизації, оскільки користувач не авторизований. Після введення необхідних даних, користувач, за допомогою браузера відправляє запит на авторизацію до Controller. У разі успішної авторизації, Controller повертає головну сторінку сайту, яку браузер відображає для користувача. Якщо авторизація не проходить, Controller повертає сторінку авторизації з інформацією про помилку.

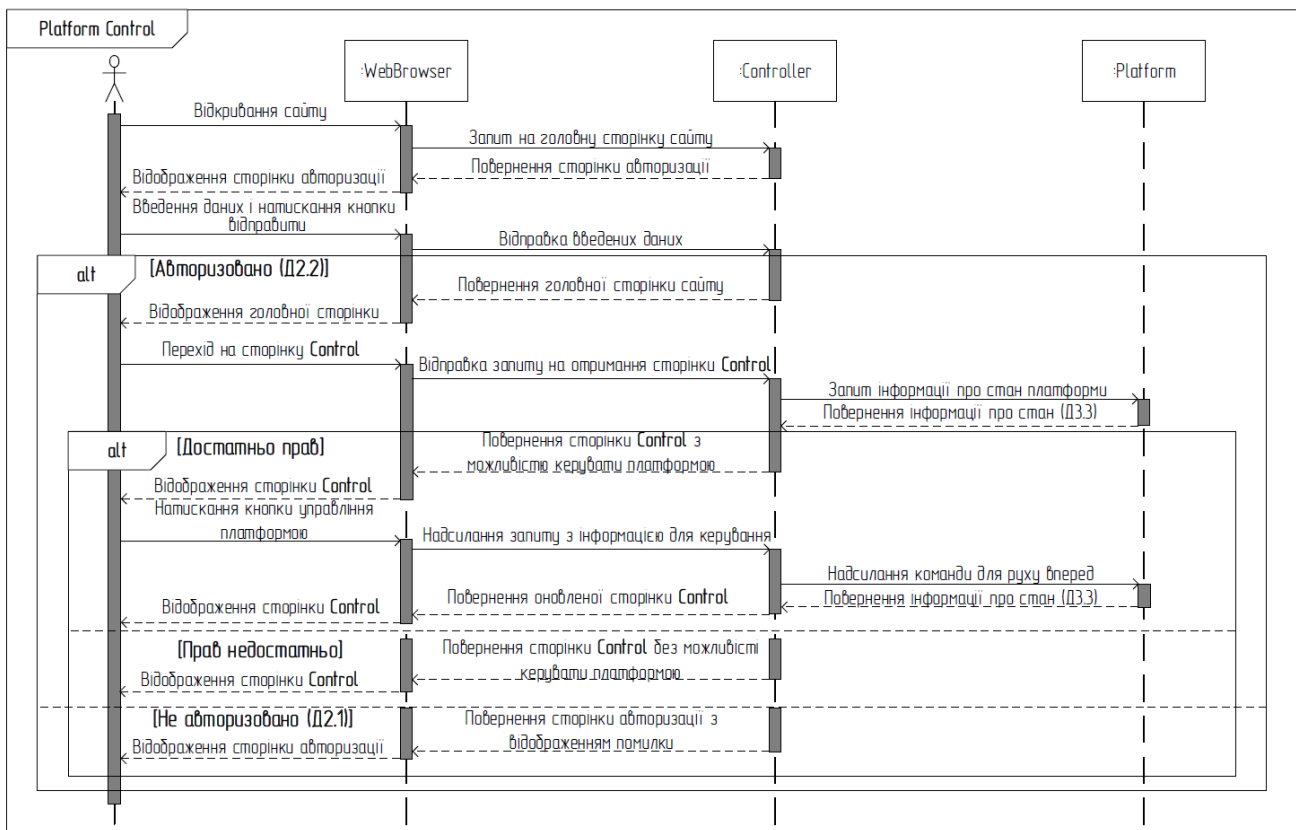


Рисунок 5.3 – Діаграма послідовності (загальна)

При переході на сторінку «Control» користувач, через браузер, надсилає запит до Controller на отримання відповідної сторінки. Controller відправляє запит до Platform для отримання інформації про її стан. Platform у відповідь повертає необхідну інформацію, яку Controller обробляє і надсилає у складі сторінки користувачу. Сторінку яку у цьому випадку повертає Controller залежить від прав користувача. Якщо користувач перебуває у ролі «Оператор» або «Адміністратор», то повертається сторінка з графічним інтерфейсом управління платформою, в іншому випадку повернеться сторінка лише з інформацією про платформу та відео.

Якщо користувач натискає кнопки управління, то браузер надсилає відповідну інформацію до Controller, який у свою чергу обробляє її і надсилає сигнали управління до Platform. НРО у свою чергу виконує отримані команди і у відповідь надсилає необхідну вихідну інформацію, яку Controller обробляє і у складі сторінки відправляє користувачу.

5.4.2 Розроблення докладної діаграми послідовності

За основу докладної діаграми послідовностей рис. 5.4 була взята загальна діаграма послідовностей, але розширена докладним описом з варіантами команд контролю.

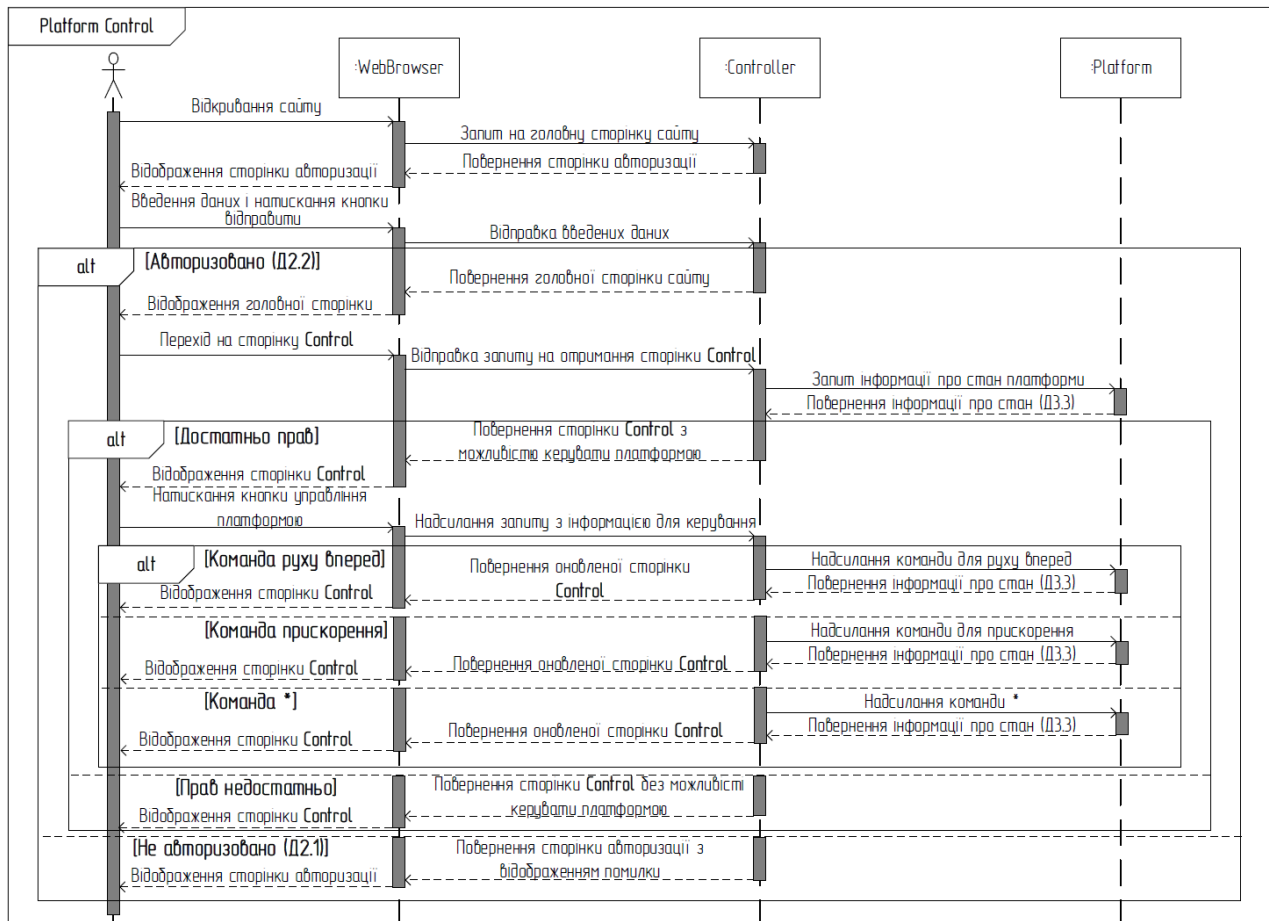


Рисунок 5.4 – Діаграма послідовності (докладна)

Отже, розглянемо основні відмінні риси докладної діаграми послідовностей.

У докладній діаграмі послідовностей наявні конкретні команди управління, які надсилаються до платформи. У прикладах, які наведені, користувач надсилає команду «Рух вперед», у відповідь на що Controller формує відповідну команду і надсилає її до платформи. Платформа виконує команду «Рух вперед» і відповідь надсилає необхідну вихідну інформацію, яку Controller обробляє і у складі сторінки відправляє користувачу.

6 РОЗРОБЛЕННЯ СТАРАПІ ПРОЕКТУ

6.1 Опис ідеї проекту

Метою розділу є розроблення програмної систем для дистанційного управління рухомими об'єктами. Розглянемо зміст опис стартап-проекту табл. 6.1, основними складовими якої є:

- зміст ідеї
- можливі напрямки застосування;
- вигоди для користувача.

Таблиця 6.1 Опис ідеї стартап-проекту

<i>Зміст ідеї</i>	<i>Напрямки застосування</i>	<i>Вигоди для користувача</i>
Програмна система для дистанційного управління рухомим об'єктом.	Віддалене управління.	Відсутність необхідності перебувати на місці.
	Дистанційний моніторинг за місцем	Оператор бачить картинку з камери рухомого об'єкту.
	Ліквідація певних видів загроз.	Відсутність необхідності ризикувати життям.

Дана архітектура відрізняється від аналогічних розширеними можливостями (наявністю сайту для віддаленого управління), відносною дешевизною, легкістю розширення.

Основними конкурентами на ринку є Starship Technologies, Eyedrive, SpyRobot. Проте у них наявні не усі функції, що пропонуються нашою системою.

Докладний огляд сильних, слабких та нейтральних характеристик інфеї проекту у порівнянні з конкурентами розглянуто на табл. 6.2

Таблиця 6.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ n/n	Техніко- економічні характеристик и ідеї	(потенційні) товари/концепції конкурентів			
		Мій проект	Starship Technologies	Eyedrive	SpyRobot
1.	Економічні	55 тис грн	80 тис грн	95 тис грн	78 тис грн
2.	Технологіч- ність	Наяв-ність сайту	Обмежена функціона- льність	Наявність багатьох функцій	Наявність багатьох функцій
3.	Надійність	Висока відказостій- кість	Висока відказостій- кість	Висока відказостій- кість	Висока відказостій- кість

Отже, до сильних сторін ідеї продукту можна віднести меншу ціну ніж у конкурентів та наявність сайту для віддаленого управління, нейтральною є надійність, оскільки вона так ж як і у конкурентів.

6.2 Технологічний аудит ідеї проекту

В межах даного підрозділу необхідно провести аудит технології, за допомогою якої можна реалізувати ідею проекту (технології створення товару).

Визначення технологічної здійсненності ідеї проекту передбачає аналіз таких складових (табл. 6.3):

Таблиця 6.3 – Технологічна здійсненність ідеї проекту

<i>№ n/n</i>	<i>Ідея проекту</i>	<i>Технології її реалізації</i>	<i>Наявність технологій</i>	<i>Доступність технологій</i>
1	Побудова сайту для віддаленого управління	Використання технологій побудови web додатків	ASP.NET Core MVC та платформи .NET	Доступна
2	Реалізація авторизації	Технологія для збереження авторизованого користувача	ASP.NET Token Based Authentication	Доступна
3	Обмін інформацією між сайтом та платформою	Вибір протоколу для обміну даними	HTTP, WebSocket, TCP	Доступна
4	Управління з використанням клавіатури	Забезпечення можливості передачі запитів при натисканні кнопок клавіатури	JavaScript, JQuery, AJAX	Доступна
Обрана технологія реалізації ідеї проекту: Проект можливо реалізувати. Сайт для віддаленого управління буде побудовано з використанням технологій ASP.NET Core MVC та платформи .NET, авторизація буде виконана на базі технології ASP.NET Token Based Authentication, обмін інформацією буде відбуватися по HTTP, управління клавіатурою буде реалізовано на базі технології JQuery, AJAX.				

Отже, за результатами аналізу таблиць можна зробити висновок, що технічна реалізація проекту можлива. В кінці таблиці було докладно розібрано технології,

що будуть використані у реалізації проекту. Вибрані технології є легкодоступними, та широко використовуються на ринку.

6.3 Аналіз ринкових можливостей запуску стартап-проекту

Для визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, побудуємо необхідні таблиці, які дозволять спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Спочатку проведемо аналіз попиту: наявність попиту, обсяг, динаміка розвитку ринку (таблиця 6.4).

Таблиця 6.4 – Попередня характеристика потенційного ринку стартап-проекту

<i>№ n/n</i>	<i>Показники стану ринку (найменування)</i>	<i>Характеристика</i>
1	Кількість головних гравців, од	4-5
2	Загальний обсяг продаж, грн/ум.од	150-200 тис за 1 систему, за рік 1000-1500 шт
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Без обмежень
5	Специфічні вимоги до стандартизації та сертифікації	Відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	20-30% за од.товару

Отже, ринок є привабливим для входженням з власною реалізацією системи дистанційного управління рухомими об'єктами, оскільки середня норма

рентабельності в галузі за одиницю товару є вищою ніж банківські відсотки на вкладення.

Далі визначимо потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи на таблиці 6.5.

Таблиця 6.5 – Характеристика потенційних клієнтів стартап-проекту

<i>Потреба, що формує ринок</i>	<i>Цільова аудиторія (цільові сегменти ринку)</i>	<i>Відмінності у поведінці різних потенційних цільових груп клієнтів</i>	<i>Вимоги споживачів до товару</i>
Створення платформ для віддаленого управління, яким потрібна змога управління через сайт	Державні спеціальні служби, групи що використовую дистанційне управління	Фактори: Наявність необхідних технологій при реалізації платформ	Надійність, безперебійна робота, коректність відображення даних, сервісне обслуговування

Після визначення потенційних груп клієнтів проведемо аналіз ринкового середовища шляхом складання таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають на таблиці 6.6 та 6.7.

Таблиця 6.6 – Фактори загроз

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1	Знецінення гривні	Зменшиться попит, в наслідок підвищення ціни	Можливе введення знижок і акцій
2	Поява нових конкурентів на ринку	Зменшення долі на ринку	Купівля реклами
3	Збільшення вартості розробки	Підвищення заробітної плати працівників	Зміна ціни продукції

Таблиця 6.7 – Фактори можливостей

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1	Підвищення популярності дистанційно керованих платформ	Збільшення попиту	Збільшення обсягів виробництва
2	Зменшення ціни розробку	Зменшення витрат	Зменшення цін зростання попиту. Вихід на нові ринки збуту

Далі проведемо аналіз пропозиції: визначимо загальні риси конкуренції на ринку табл. 6.8.

Таблиця 6.8. Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)</i>
1. Вказати тип конкуренції - олігополія	Невелика кількість виробників, переважно нецінова конкуренція	Впровадження реклами, гарантування якості продукції, реалізація сервісного обслуговування
2. За рівнем конкурентної боротьби - національний	Конкурентна боротьба в межах країни	Вихід на світовий ринок
3. За галузевою ознакою міжгалузева	Боротьба між товариствами різних галузей	Реалізація сайту для дистанційного управління, створення додаткових індивідуальних функцій
4. Конкуренція за видами товарів: - товарно-видова	Види однієї категорії товарів, що можуть задовільнити конкретну потребу споживача	Створення додаткових функцій для конкретних потреб
5. За характером конкурентних переваг - цінова	Використанням цін як засобу досягнення кращих економічних умов збуту і ринку	Організація відповідної цінової політики
6. За інтенсивністю - не марочна	Відсутня прив'язаність до певної марки, можливість співпраці з будь-ким	Співпраця з різними марками

Після аналізу конкуренції проведемо більш детальний аналіз умов конкуренції в галузі на таблиці 6.9.

Таблиця 6.9 – Аналіз конкуренції в галузі за М. Портером

	<i>Прямі конкуренти в галузі</i>	<i>Потенційні конкуренти</i>	<i>Постачальники</i>	<i>Клієнти</i>	<i>Товари-замінники</i>
<i>Складові аналізу</i>	Starship Technologies, Eyedrive, SpyRobot	Відсутні	Відсутні	Контроль якості Чутливість до зміни цін	Ціна Змінні витрати
Висновки:	Інтенсивність конкурентності низька – ринок досить широкий	можливість входу на ринок є; потенційні конкуренти відсутні; можливий вихід на ринок на протязі до 1 року;	Постачальники відсутні, впливу на умови роботи на ринку немає.	Клієнти диктують умови. Вимоги, в основному висовуються до якості та ціни	Обмеження полягають у ціновій політиці на товари заміни.

Робота на ринку можлива, тому що відсутні потенційні конкуренти, а динаміка ринку відображає потребу у подібних товарах. Перевагами системи є наявність сайту для віддаленого управління, що поставляється у комплекті.

На основі аналізу конкуренції, проведеного в таблиці 6.9, а також із урахуванням характеристик ідеї проекту (табл. 6.2), вимог споживачів до товару (таблиці 6.6) та факторів маркетингового середовища (таблиці 6.7-6.8) визначимо

та обґрунтуємо перелік факторів конкурентоспроможності. Аналіз оформлюється за табл. 6.10

Таблиця 6.10 – Обґрунтування факторів конкурентоспроможності

№ n/n	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Ціна	Чутливість споживача до цін
2	Якість	Важливим є коректність передачі і відображення інформації
3	Довговічність	Відсутність потреби у заміні системи
4	Масштабованість	Можливість замовлення різних варіантів системи

За визначеними факторами конкурентоспроможності (табл. 6.10) проводиться аналіз сильних та слабких сторін стартап-проекту (табл. 6.11).

Таблиця 6.11 – Порівняльний аналіз сильних та слабких сторін «назва проекту»

№ n/ n	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з SpyRobot						
			-3	-2	-1	0	+1	+2	+3
1	Ціна	20	+						
2	Якість	11					+		
3	Довговічність	10				+			
4	Масштабованість	17		+					

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak)

сторін, загроз (Troubles) та можливостей (Opportunities) (табл. 6.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін (табл. 6.11).

Перелік ринкових загроз та ринкових можливостей складемо на основі аналізу факторів загроз та факторів можливостей маркетингового середовища. Ринкові загрози та ринкові можливості є наслідками (прогнозованими результатами) впливу факторів, і, на відміну від них, ще не є реалізованими на ринку та мають певну ймовірність здійснення.

Таблиця 6.12 – SWOT- аналіз стартап-проекту

Сильні сторони: Наявність додаткового функціоналу, нища ціна ніж у конкурентів.	Слабкі сторони: Менш надійний ніж деякі конкуренти
Можливості: Збільшення попиту на товар, зменшення ціни на розробку	Загрози: Поява нових конкурентів, зменшення попиту на товар

На основі SWOT-аналізу розробимо альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок (див. табл. 6.9, аналіз потенційних конкурентів).

Визначені альтернативи проаналізуємо з точки зору строків та ймовірності отримання ресурсів (табл. 6.13).

В якості обраної альтернативи буде виступати друга та четверта. Розроблення додаткових функцій є досить перспективною, та строки реалізації достатньо короткі.

Таблиця 6.13 – Альтернативи ринкового впровадження стартап-проекту

<i>№ п/п</i>	<i>Альтернатива (орієнтовний комплекс заходів) ринкової поведінки</i>	<i>Ймовірність отримання ресурсів</i>	<i>Строки реалізації</i>
1	Здешевлення ціни на розробку	Низька	До одного року
2	Розроблення додаткових функцій	Висока	До двох місяців
3	Підвищення продуктивності	Висока	До шести місяців
4	Запуск реклами	Середня	До одного місяця

6.4 Розроблення ринкової стратегії проекту

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів (табл. 6.14).

Таблиця 6.14 – Вибір цільових груп потенційних споживачів

<i>№ п/п</i>	<i>Опис профілю цільової групи потенційних клієнтів</i>	<i>Готовність споживачів сприйняти продукт</i>	<i>Орієнтовний попит в межах цільової групи (сегменту)</i>	<i>Інтенсивність конкуренції в сегменті</i>	<i>Простота входу у сегмент</i>
1	Військова галузь	так	10%	Висока	Складна

Продовження таблиці 6.14

2	Комунальні служби	так	20%	Висока	Середня
3	Державні установи	так	10%	Висока	Складна
4	Роздрібні споживачі	так	60%	Середня	Легка
У якості цільових груп обрано: військова галузь, комунальні служби та роздрібні споживачі					

За результатами аналізу потенційних груп споживачів (сегментів) автори ідеї обирають цільові групи, для яких вони пропонуватимуть свій товар, та визначають стратегію охоплення ринку:

- якщо компанія зосереджується на одному сегменті – вона обирає стратегію концентрованого маркетингу;
- якщо працює із кількома сегментами, розробляючи для них окремо програми ринкового впливу – вона використовує стратегію диференційованого маркетингу;
- якщо компанія працює із всім ринком, пропонуючи стандартизовану програму (включно із характеристиками товару/послуги) – вона використовує масовий маркетинг.

Для роботи в обраних сегментах ринку необхідно сформувати базову стратегію розвитку (табл. 6.15).

У якості альтернативи розвитку оберемо стратегію виклику лідера, стратегією охоплення ринку буде стратегія диференційного маркетингу.

Таблиця 6.15 – Визначення базової стратегії розвитку

<i>Обрана альтернатива розвитку проекту</i>	<i>Стратегія охоплення ринку</i>	<i>Ключові конкурентоспроможні позиції відповідно до обраної альтернативи</i>	<i>Базова стратегія розвитку</i>
Стратегія виклику лідера	Стратегія диференційного маркетингу	Індивідуальний підхід до клієнта; кращий набір функцій і ціна ніж у конкурента	Стратегія диференціації передбачає надання товару важливих з точки зору споживача відмінних властивостей, які роблять товар відмінним від товарів можливих конкурентів.

Наступним кроком виберемо стратегію конкурентної поведінки (табл. 6.16).

На основі вимог споживачів з обраних сегментів до постачальника (стартап-компанії) та до продукту (див. табл. 6.6), а також в залежності від обраної базової стратегії розвитку (табл. 6.15) та стратегії конкурентної поведінки (табл. 6.16) розробляється стратегія позиціонування (табл. 6.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку/проект.

Таблиця 6.16 – Визначення базової стратегії конкурентної поведінки

<i>Чи є проект «першопрохідцем» на ринку?</i>	<i>Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?</i>	<i>Чи буде компанія копіювати основні характеристики товару конкурента, і які?</i>	<i>Стратегія конкурентної поведінки</i>
Ні	Буде відбуватись пошук нових та збирання існуючих споживачів	Стандартний набір буде скопійовано, але додано багато нововведень	Стратегія виклику лідера

Таблиця 6.17 – Визначення стратегії позиціонування

<i>Вимоги до товару цільової аудиторії</i>	<i>Базова стратегія розвитку</i>	<i>Ключові конкурентоспромо жні позиції власного стартап- проекту</i>	<i>Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)</i>
Низька ціна, надійність, функціональність	Стратегія диференц іювання	-Наявність сайту для віддаленого керування, розширюваність	Постійне оновлення(вдосконал ення) Найкраща якість Розширюваність

Отже, у якості цільових груп було обрано військову галузь, комунальні служби та роздрібні споживачі. У якості стратегії для охоплення ринку було обрано стратегію диференційного маркетингу. Стратегія конкурентної поведінки була вибрана стратегія виклику лідера.

6.5 Розроблення маркетингової програми стартап-проекту

Першим кроком є формування маркетингової концепції товару, який отримає споживач. Для цього у таблиці 6.18 потрібно підсуємо результати попереднього аналізу конкурентоспроможності товару.

Таблиця 6.18 – Визначення ключових переваг концепції потенційного товару

<i>№ n/n</i>	<i>Потреба</i>	<i>Вигода, яку пропонує товар</i>	<i>Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)</i>
1	Віддалене управління	Наявність додаткових функцій для віддаленого управління	Реалізовано сайт для віддаленого управління
1	Масштабованість системи	Просте додавання нових елементів системи	Забезпечена можливість додавання функцій, спеціально для кожного з клієнтів
1	Відображення відео з камери рухомого об'єкта	Надає можливість переглядати відео з камери	Відео можна переглядати на сайті для віддаленого управління
1	Низька ціна	Конкурентна ціна	Ціна нижча ніж у більшості конкурентів

Далі розробимо трирівневу маркетингову модель товару: уточнимо ідею продукту та/або послуги, її фізичні складові, особливості процесу її надання (табл. 6.19). В таблиці М/Нм – монотонні/немонотонні; Вр/Тх/Тл/Е/Ор – вартісні/технічні/ технологічні/ ергономічні/ органолептичні;

Після формування маркетингової моделі товару було особливо відмічено – чим саме проект буде захищено від копіювання. Захист може бути організовано за рахунок захисту ідеї товару (захист інтелектуальної власності), або ноу-хау, чи комплексне

поєднання властивостей і характеристик, закладене на другому та третьому рівнях товару.

Опис трьох рівнів моделі товару – один з найважливіших елементів розробки стартап проекту, оскільки описується загальна концепція товару.

Таблиця 6.19 – Опис трьох рівнів моделі товару

Рівні товару		Сутність та складові		
I.Товар за задумом		Система для дистанційного управління рухомим об’єктом		
II.Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор	
	1.Інтерфейс для управління на сайті	М	Тх/Тл/Е	
	2.Універсальність для різних РО	Нм	Вр/Тх/Тл	
	3.Частота надсилання інформації	М	Тх/Тл	
	4. Якість переданого відео	М	Вр/Тх/Тл	
	5. Необмежений термін дії	М	Вр/Тл	
	Якість: відповідає нормам розробки програмного забезпечення.			
	Пакування: Встановлення системи спеціалістами фірми виробника			
	Марка: ПП «Віддалуп»			
За рахунок чого потенційний товар буде захищено від копіювання: Товар буде запатентований				

Наступним кроком є визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар (остаточне визначення ціни відбувається під час фінансово-економічного аналізу проекту), яке передбачає аналіз ціни на товари-

аналоги або товари субститути, а також аналіз рівня доходів цільової групи споживачів (табл. 6.20). Аналіз проводиться експертним методом.

Таблиця 6.20 – Визначення меж встановлення ціни

<i>Рівень цін на товари-замінники</i>	<i>Рівень цін на товари-аналоги</i>	<i>Рівень доходів цільової групи споживачів</i>	<i>Верхня та нижня межі встановлення ціни на товар/послугу</i>
Відсутні	85 тис. грн	900 тис. грн	40 тис. грн – 70 тис. грн

Наступним кроком визначимо оптимальну систему збуту, в межах якого приймемо рішення (табл. 6.21):

- проводити збут власними силами або залучати сторонніх посередників (власна або залучена система збуту);
- вибір та обґрунтування оптимальної глибини каналу збуту;
- вибір та обґрунтування виду посередників.

Таблиця 6.21 – Формування системи збуту

<i>Специфіка закупівельної поведінки цільових клієнтів</i>	<i>Функції збуту, які має виконувати постачальник товару</i>	<i>Глибина каналу збуту</i>	<i>Оптимальна система збуту</i>
Купування у роздріб	Встановлення системи спеціалістами компанії виробника	Виробник - споживач	Web-сайт

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (табл. 6.22).

Концепція маркетингових комунікацій яскраво відображає специфіку поведінки цільових клієнтів, канали комунікації, якими вони користуються, завдання для рекламного повідомлення та концепцію рекламного звернення.

Таблиця 6.22 – Концепція маркетингових комунікацій

<i>Специфіка поведінки цільових клієнтів</i>	<i>Канали комунікацій, якими користуються цільові клієнти</i>	<i>Ключові позиції, обрані для позиціонування</i>	<i>Завдання рекламного повідомлення</i>	<i>Концепція рекламного звернення</i>
Ретельний вибір товару, порівняння з аналогами	Ділові зустрічі, телефони, електронна пошта	Підтримка, індивідуальний підхід, можливість розширення	Донесення переваг до клієнтів	Віддалене управління РО через сайт – це легко

З першого етапу аналізу, можна зробити висновок що розроблена система має багато сильних сторін у порівнянні з конкурентами, а по інших вона аналогічна. Проведений на другому етапі технологічний аудит показав, що усі необхідні для виробництва технології наявні і легко доступні.

Ринкові можливості запуску стартап-проекту, досліджені на третьому етапі показали доцільність виводу продукції по даній ідеї на ринок. Крім того, були розглянуті можливі ризики виходу на ринок та аналіз конкуренції, проте вони не значні.

Попередня ринкова стратегія проекту була розроблена на четвертому етапі, після чого, на п'ятому етапі, була розроблена маркетингова програма.

Тому можна зробити висновок, що програмна система дистанційного управління рухомими об'єктами є перспективною для впровадження її у виробництво.

ВИСНОВОК

У даній дисертації був розроблений макет для загальної архітектури програмних комплексів для дистанційного управління. Була проведена формалізація ПКДУ, та розглянуто основні моменти, через що вона ускладнена. Розроблені необхідні діаграми для відображення основних аспектів ПКДУ. Основний акцент, при розробці ПКДУ, був зроблений на розроблення сайту для віддаленого управління.

Рухливі роботи застосовуються переважно в екстремальних умовах, коли люди не можуть перебувати всередині транспортного засобу або якщо їх перебування там пов'язане з ризиком для життя. Крім того, в умовах підвищеної небезпеки людина починає робити помилки, знижується його працездатність і ефективність дій.

У ряді випадків доцільно використовувати роботи, дистанційно керовані оператором або групою операторів по теле- і радіоканалах зв'язку. Однак майже завжди необхідно забезпечити короткочасне, але повністю автономне функціонування робота (в першу чергу його системи управління рухом) на випадок раптового припинення радіозв'язку при входженні в зону відсутності сигналу. Крім того, телеуправління транспортним роботом вимагає від оператора постійної уваги, що вельми втомлює. У деяких робочих середовищах (зокрема, в тумані, в диму, вночі), де оператори часто втрачають орієнтування.

СПИСОК ЛІТЕРАТУРИ

1. OMG Unified Modeling Language [Електронний ресурс] – режим доступу до ресурсу: <https://www.omg.org/spec/UML/2.5.1> (дата звернення 31.05.2018). – Назва з екрану.
2. Nilesh J. Uke, Ravindra C. Thool. Objects tracking in video: a object-oriented approach using Unified Modeling Language. International Journal of Computational Vision and Robotics (IJCVR), Vol. 5, No. 2, 2015.
3. Infantino, I., Cossentino, M. and Chella, A. An agent based multilevel architecture for robotics vision systems', Proceedings of the International Conference on Artificial Intelligence, IC-AI Las Vegas, pp.386–390, 2002.
4. Shi-Xiang, T. and Wang, S. The conceptual design and simulation of mechatronic system base on UML, 2010 2nd International Conference on Computer Engineering and Technology, pp.V6-188–V6-192, 2010.
5. Gavrilescu, M. Towards UML software models for cyber physical system applications, 2012 20th Telecommunications Forum (TELFOR), No. 2, pp.1701–1704, 2012.
6. Evans, A.S. and Wellings, A.J. UML and the formal development of safety-critical real-time systems, Applicable Modelling, Verification and Analysis Techniques for Real-Time Systems, IEE Colloquium on, Vol. 2, No. 1, pp.2–5, 1999.
7. Yuan, L., Tang, T. and Liu, Y. Research on third-party test bench of train control system using UML, 2011 IEEE International Conference on Service Operations, Logistics, and Informatics (SOLI), pp.560–564, 2011.
8. Наслідки аварії ЧАЕС [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://bluesbag6.narod.ru/index41.html> (дата звернення 01.05.2018) – Назва з екрана.
9. Наслідки аварії на АЕС Фукусіма - 1 [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу http://rb.mchs.gov.ru/mchs/radiation_accidents/fukushima_daiichi_accident/item/8914 (дата звернення 01.05.2018) – Назва з екрана.

10. Starship [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу www.starship.xyz (дата звернення 01.05.2018) – Назва з екрана.
11. Dispatch [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://www.simplebotics.com/2016/04/dispatch-joins-deliver-things-robot.html> (дата звернення 01.05.2018) – Назва з екрана.
12. Riley [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://robotrends.ru/pub/1631/vstrechayte---riley-domashniy-patrulnyy-robot> (дата звернення 01.05.2018) – Назва з екрана.
13. SpyRobot 4WD [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу http://defense-update.com/20070501_spyrobot4wd.html (дата звернення 01.05.2018) – Назва з екрана.
14. SpyRobot 4WD 6x6 [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <https://newatlas.com/the-spyrobot-6x6-new-back-packable-unstoppable-ugv/9030/> (дата звернення 01.05.2018) – Назва з екрана.
15. Автоматизована рухома платформа Toshiba [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://chornobyl.in.ua/robot-dlya-dezaktivacii.html> (дата звернення 01.05.2018) – Назва з екрана.
16. qinetiq North America [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу www.qinetiq-na.com (дата звернення 17.12.2017) – Назва з екрана.
17. Precision Urban Hopper [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <https://www.dailytechinfo.org/military/578-precision-urban-hopper-voennyj-robot-prygayushhiy.html> (дата звернення 01.05.2018) – Назва з екрана.
18. Краулеры (ползающие роботы) [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://robotrends.ru/robopedia/kraulery-polzayushie-roboty> (дата звернення 01.05.2018) – Назва з екрана.
19. Роботы в сфере энергетики (ползающие роботы) [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://vse->

elektrichestvo.ru/novosti/roboty-v-sfere-energetiki.html (дата звернення 01.05.2018) – Назва з екрана.

20. Wicron [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <https://wicron.com> (дата звернення 01.05.2018) – Назва з екрана.

21. Raspberry pi wifi radio controlled rc vehicle robot [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <https://custom-build-robots.com/raspberry-pi-car-junior/raspberry-pi-robot-steering-internet-smartphone/6961?lang=en#prettyPhoto> (дата звернення 01.05.2018) – Назва з екрана.

22. UML examples and algorithms [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://www.omg-portal.ru/article/material17> (дата звернення 01.05.2018) – Назва з екрана.

23. ASP.NET MVC Overview [Електронний ресурс] — режим доступу до ресурсу: [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx) (дата звернення 31.05.2018). — Назва з екрану.

24. Роб Камерон, Дэйл Михалк, ASP.NET 3.5, компоненты AJAX и серверные элементы управления для профессионалов, 560 ст., 2010 р.

25. Адам Фримен, ASP.NET MVC 5 с примерами на C# 5.0 для профессионалов 5-е издание, 736 ст., 2015 р.

26. Короткі відомості про бази даних та системи управління базами даних [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <https://studfiles.net/preview/5607354/> (дата звернення 01.05.2018) – Назва з екрана.

27. ДСТУ Ергономіка [Електронний ресурс] : [Веб-сайт]. – Електронні дані. – Режим доступу <http://nau.edu.ua/site/variables/docs/science/ndi/dizain/ДСТУ-РМ%20сидячи.doc> (дата звернення 01.05.2018) – Назва з екрана.

28. Introduction to Electronic Engineering and Computer Science, MIT, p.117-168, 2011.

ДОДАТОК А Лістинги основних класів сайту для віддаленого управління

```

public class FileHelper
{
    private static readonly string _infoPath = @"D:\information.txt";
    private static readonly string _controlPath = @"D:\control.txt";
    public static InfoFile ReadInfo()
    {
        InfoFile info = new InfoFile();
        using (StreamReader reader = new StreamReader(_infoPath, true))
        {
            try
            {
                var readed = reader.ReadLine().Split('|');
                info.Operation = int.Parse(readed[0]);
                info.SpeedChange = int.Parse(readed[1]);
                info.Speed = int.Parse(readed[2]);
                info.LocX = int.Parse(readed[3]);
                info.LocY = int.Parse(readed[4]);
                info.Status = int.Parse(readed[5]);
            }
            catch(Exception ex)
            {
                info.ErrorCode = 100;
                info.ErrorMessage = ex.Message;
            }
        }
        return info;
    }
    public static ControlFile ReadControl()
    {
        ControlFile control = new ControlFile();
        using (StreamReader reader = new StreamReader(_controlPath, true))
        {
            try
            {
                var readed = reader.ReadLine().Split('|');
                control.Operation = int.Parse(readed[0]);
                control.SpeedChange = int.Parse(readed[1]);
            }
            catch (Exception ex)
            {
                control.ErrorCode = 100;
                control.ErrorMessage = ex.Message;
            }
        }
    }
}

```

```

        }
    }
    return control;
}
public static void SetControl(ControlFile control)
{
    using (StreamWriter file = new StreamWriter(_controlPath, false))
    {
        file.WriteLine(control.ToString());
    }
}
public static void SetInfo(InfoFile info)
{
    using (StreamWriter file = new StreamWriter(_infoPath, false))
    {
        file.WriteLine(info.ToString());
    }
}
}
public class HomeController : Controller
{
    [HttpPost]
    public string MoveControl([FromBody]string operation)
    {
        var current = FileHelper.ReadControl();
        current.Operation = int.Parse(operation);
        FileHelper.SetControl(current);
        return "good";
    }
    [Authorize()]
    public void SpeedControl([FromBody]int operation)
    {
        var current = FileHelper.ReadControl();
        current.SpeedChange = operation;
        FileHelper.SetControl(current);
    }
    [Authorize()]
    public ActionResult Index()
    {
        return View("Main");
    }
    [Authorize()]
    public ActionResult About()
    {
        return View();
    }
}

```

```

    }
    [Authorize()]
    public ActionResult Guide()
    {
        return View();
    }
    [Authorize()]
    public ActionResult PlatformControll()
    {
        return View();
    }
    [Authorize()]
    public async Task<IActionResult> Logout()
    {
        await HttpContext.SignOutAsync();
        return RedirectToAction("Index", "Accounting");
    }

    [HttpPost]
    public ControlFile GetControl([FromBody]InfoFile info)
    {
        FileHelper.SetInfo(info);
        return FileHelper.ReadControl();
    }
    [HttpGet]
    public InfoFile GetInitInfo()
    {
        return FileHelper.ReadInfo();
    }
    [HttpGet]
    public ControlFile GetInitControl()
    {
        return FileHelper.ReadControl();
    }
    [HttpGet]
    public InfoFile GetInfo()
    {
        return FileHelper.ReadInfo();
    }
}

```

ДОДАТОК Б Публікації